Updated mobility analysis
Through MZ2015

Working Paper

Author(s):
Smith, Abigail

Publication date:
2018-01

Permanent link:
https://doi.org/10.3929/ethz-b-000213407

Rights / license:
In Copyright - Non-Commercial Use Permitted

Originally published in:
Arbeitsberichte Verkehrs- und Raumplanung 1309
Updated Mobility Analysis: Through MZ2015

Abigail Smith

Arbeitsbericht Verkehrs- und Raumplanung 1309
January 2018
Abstract

In the following document is an explanation of the methodology behind the calculation of the updated mobility graphs. It also includes any remaining concerns or comments with respect to the data. Swiss travel diaries dating from 1994 to 2015 are used to plot the evolution of the travel patterns amongst different ages, genders and income terciles. Understanding the developments in travel patterns is the key to implementing the most effective transportation systems.

Preferred citation style

# Table of contents

1  Introduction to Mobility Patterns ................................................................. 6

2  Analysis by Gender ......................................................................................... 6
   2.1  Driver’s License Ownership ................................................................. 7
   2.2  Share of Car Availability ....................................................................... 7
   2.3  Share of any public transportation subscription ................................. 8
   2.4  Share of a public transportation season ticket ...................................... 9
   2.5  Number of Daily Trips ........................................................................... 10
   2.6  Daily Travel Distance ........................................................................... 12
   2.7  Daily Travel Time .................................................................................. 12
   2.8  Average Travel Speed ........................................................................... 13
   2.9  Average Travel Speed when Car Always Available ......................... 14
   2.10 Average Travel Speed on Car Trips ..................................................... 15

3  Analysis by Mobility Tool Ownership .......................................................... 16
   3.1  Ownership of Public Transportation Subscriptions ........................... 17
   3.2  Chosen Trip Method ............................................................................. 18
   3.3  Daily Distance Travelled ....................................................................... 20
   3.4  Number of Trips per Day ...................................................................... 22
   3.5  Sum of Daily Travel Time ..................................................................... 24
   3.6  Average Travel Speed .......................................................................... 26

4  Analysis by Income Terciles .......................................................................... 28
   4.1  License Ownership ................................................................................. 29
   4.2  Car Availability ...................................................................................... 29
   4.3  Public Transportation Subscription ....................................................... 30
   4.4  Daily Distance Travelled ....................................................................... 31
   4.5  Average Number of Trips ..................................................................... 33
   4.6  Daily Travel Time .................................................................................. 34
   4.7  Average Travel Speed .......................................................................... 34

5  Introduction to Activity Chain Analysis ......................................................... 35
5.1 Home and Out-of-Home Activities by Time ........................................... 36
5.2 Work Time ......................................................................................... 38
5.3 Business Activity Time ................................................................. 39
5.4 Leisure Activity Time ....................................................................... 41
5.5 Time in Education ............................................................................ 42
5.6 Shopping Time .................................................................................. 44
5.7 Time in Other Activities ................................................................. 45
5.8 Out-of-Home Time split by Activity ............................................... 47
5.9 Number of Journeys .......................................................................... 47
5.10 Distance of Journeys for Activities ................................................. 48

6 Reference list .................................................................................... 49
A SPSS Scripts .................................................................................. 50
  A.1 Mobility Analysis 2000 ............................................................... 50
  A.2 Mobility Analysis 2005 ............................................................... 59
  A.3 Mobility Analysis 2010 ............................................................... 64
  A.4 Mobility Analysis 2015 ............................................................... 73
  A.5 Activity Chain Analysis 2005 ...................................................... 83
  A.6 Activity Chain Analysis 2010 ...................................................... 90
  A.7 Activity Chain Analysis 2015 ...................................................... 97

List of tables
Table 1: Activity Grouping ...................................................................... 36

List of figures
Figure 1: Driver’s License Ownership .................................................. 7
Figure 2: Share of Car Availability ...................................................... 8
Figure 3: Share of any PT subscription ............................................... 9
Figure 4: Share of a GA or season ticket for public transportation ......... 10
Figure 5: Number of daily trips ......................................................... 11
Figure 6: Daily travel distance

Figure 7: Number of daily trips

Figure 8: Average travel speed

Figure 9: Average car speed when car available

Figure 10: Average speed of car trips

Figure 11: Mobility tool ownership separated by gender

Figure 12: Mobility tool ownership comparison with 2010

Figure 13: Trips per day by mobility tool ownership

Figure 14: Trips per day by mobility tool ownership and non-motorised transport

Figure 15: Sum of daily distance travelled

Figure 16: Sum of daily distance travelled

Figure 17: Sum of daily distance travelled

Figure 18: Average travel speed by mobility tool

Figure 19: Average travel speed by mobility tool

Figure 20: License ownership by income tercile

Figure 21: Car availability by income tercile

Figure 22: PT subscription by income tercile

Figure 23: Distance travelled by income tercile

Figure 24: Number of trips by income tercile

Figure 25: Travel time by income tercile

Figure 26: Travel speed by income tercile

Figure 27: Time at home and out-of-home activities

Figure 28: Time at home and out-of-home activities by gender

Figure 29: Time spent at work

Figure 30: Time spent at work by gender

Figure 31: Time spent at business activities

Figure 32: Time spent at business activities by gender

Figure 33: Time spent at leisure activities

Figure 34: Time spent at leisure activities by gender

Figure 35: Time spent in education

Figure 36: Time spent in education by gender

Figure 37: Time spent shopping

Figure 38: Time spent shopping by gender

Figure 39: Time spent in other activities

Figure 40: Time spent in other activities by gender

Figure 41: Time spent in other activities

Figure 42: Summary of number of journeys for all activity types
Figure 43: Summary of average distance travelled for all activity types........................................ 48
1 Introduction to Mobility Patterns

The analysis on the mobility patterns in Switzerland was originally conducted in “Mobility patterns in Switzerland: past, present and future” by Ciari et al. (2013). The subsequent analysis documented in this paper follows the procedures implemented in the original paper as closely as possible, whilst documenting the process for future use. For most of the graphs depicted below, the data for 2010 and earlier was processed by Ciari et al. (2013).

The households and target persons within the micro-censuses are separated into cohorts based upon year of birth. These cohorts are the basis of analysis for each of the graphs. For 2005, 2010 and 2015, the age limits of the cohorts are determined by the year of the census minus the boundary years of the cohort minus an additional 1. For example someone who is 43 years old in the 2015 micro-census would fall into the 1970-79 age cohort because the upper and lower limits for this cohort are 44 and 35 respectively (2015-1970-1 = 44). The additional minus one accounts for the fact that the census is taken at the start of the year and most people are born after January. For the years 2000 and earlier, the age cohort brackets are simply calculated as the census year minus the cohort year. Since the precise reasons for the switch in methodology are unknown, the data is kept in its original format and 2015 follows the precedent of 2010.

The SPSS Syntax scripts used for all of the analyses in this paper are included in the appendices. Similarly, the Excel file containing the raw data for the graph collected from SPSS is also contained within the appendix.

2 Analysis by Gender

The first analysis of the micro-census data is completed with respect to age and gender. Despite the increase in equality between men and women politically and socially, there are still evident differences between the genders with regards to their travel patterns. Knowing and understanding these differences aids transportation planners in providing the best services for the entire population. For each of these graphs only data from the 2015 micro-census is added as the previous data was already collected and processed.
2.1 Driver’s License Ownership

The added values from the 2015 micro-census follow the existing trends with high license ownership for the majority of the population. There are only 12 system missing values in MZ2015 for this item.

Figure 1: Driver’s License Ownership

![Driver’s License Ownership Graph](image)

2.2 Share of Car Availability

The car availability values for the year 2000 and before appear to be calculated in a manner relating to the ratio of cars per license in the household, though the precise calculation method is unknown. For 2005 and onwards, the values provided come from the car availability question. These values are adjusted so that the car is only considered available when it is always available and is not available for all other responses (including most system missing values which were recoded to 0 only when a driver’s license was not owned). There are 131 system missing values in MZ2015 for this item. The discrepancy in calculation method would explain the jump between 2000 and 2005 which is particularly visible in the male split.
2.3 Share of any public transportation subscription

Since the previous micro-censuses do not have the same level of specificity as the recent ones with regards to public transportation subscriptions, this percentage is the ownership of any public transportation subscription or discount card. For the 2015 micro-census, this includes Halbtax and junior cards as well as the general, regional and route subscriptions. In past micro-censuses, the PT subscription was documented as annually, monthly, weekly, Halbtax and other. The inclusion of the Halbtax makes this an inaccurate description of true commitment to public transport, as the Halbtax does not allow the individual to travel with public transport for a flat rate but instead offers a discount on purchased tickets. It skews the data to show more people possessing subscriptions than actually do. Due to the method of calculating this term for 2015, there are no system missing values as those values are coded as 0.
2.4 Share of a public transportation season ticket

Unlike the previous graph, an individual is only considered to possess a public transportation subscription if they have a general, regional or route subscription. As the censuses prior to 2000 do not contain the required specificity, they are excluded from the analysis. The ownership of public transportation season tickets indicate a commitment by the individual to travelling via public transportation. Unlike most of the other charts displayed, this one shows little difference between genders.
2.5 Number of Daily Trips

The number of daily trips is limited to only trips within Switzerland (Inland). In MZ2015, the counts of daily trips range from 0 to 21 with the largest percentage of individuals having only two trips. There are no missing values.
Figure 5: Number of daily trips
2.6 **Daily Travel Distance**

Daily travel distance is the sum of the domestic distances travelled. The distances range from 0 to 986 with no missing values for the MZ 2015 data.

Figure 6: Daily travel distance

![Graph showing daily travel distance by cohort age.](image)

2.7 **Daily Travel Time**

Daily travel time is a plot of the average total time (in minutes) spent travelling, including waiting time. As with the distance, this is also only for domestic travel. The MZ2015 values range from 0 to 1245 with an average of 92 minutes and no missing values.
2.8 Average Travel Speed

The average travel speed is calculated by dividing the total daily travel time by the total daily travel time (including waiting time). A clear division is visible in the graph between men and women. This variable has significant amounts of missing values (approximately 11-12% in each cohort). Only the Mikrozensi 2000 – 2015 are included in the analysis.
2.9 Average Travel Speed when Car Always Available

The travel speed in this case is restricted to only those individuals who have a car always available to them. Through this limit, it is assumed that a majority of the trips are taken with a car. On average, there is a clear distinction between the driving speed achieved by men and women given their movement patterns. Particularly in the younger age cohorts, the males drive between five and ten km/hr faster than women. For the older age cohorts, the difference is less visible but is still clearly there, indicating less variation in patterns of movement at these ages.
2.10 Average Travel Speed on Car Trips

The average travel speed by car trips only is calculated using data from the domestic trips file. First the cases are constrained to those for which a car is the mode of transportation. Then, the speed for each section of the trip is calculated by dividing the “on the road” time by the routed distance for the section. An average of these values is taken and prescribed to each unique household and target person combination. Following that, the age cohort analysis is added to the data and the analysis is completed. Only the 2010 and 2015 micro censuses have enough information to calculate this trip time accurately, and therefore, the travel speed given car availability is best for seeing change over time. However, the findings visible in this graph corroborate the gender speed gap and serve to validate the data in the previous graphs. Whilst the speed difference is smaller than the difference in the car availability graph, it remains in the range of three to six km/hr.
3 Analysis by Mobility Tool Ownership

The percentages of the population that own each of the different mobility tools are depicted in the graph below. Only the availability of a car and a public transportation subscription are considered, though there is data available for bike and motorbike ownership as well. For a car to be marked as available, it must be available all of the time. Thus, there are several instances where individuals will have access to a car if it is requested or the individuals share a car which are coded as not having a car available at all. Additionally, a public transportation subscription does not include the Halbtax or other discount cards as it only includes general, regional and route subscriptions.

In 2010 and 2015, the public transportation data is separated into these distinct categories; however, in the earlier micro-censuses, the data is categorized as a weekly, monthly or annual subscription. That discrepancy in the data could account for the slight differences seen in the distribution.

In order to reduce the number of system missing values for this analysis, any individual who does not own a driver’s license is marked as also not having a car available. The remaining system missing values correspond to individuals who have a license but did not give a response on car availability. They make up a small percentage of the total group (0.3%).
The split between male and female individuals illustrates how men make up a higher percentage of car ownership while women tend to possess more public transportation subscriptions.

Figure 11: Mobility tool ownership separated by gender

The SPSS Syntax scripts for the mobility tool ownership are included within the scripts for the gender-based analysis.

### 3.1 Ownership of Public Transportation Subscriptions

The ownership of public transportation subscriptions by car availability is remarkably similar to the data found in 2010. With a visual comparison of the two graphs it is next to impossible to identify a difference between them, and while the percentages to vary slightly, in many instances it is no more than a few decimal points. As usual, the data is weighted with the person weight. Here any subscription under ‘other’ or ‘Strecken’ are counted as not having a subscription, and the GA category covers anyone who has the GA card even if they also have another subscription (144 individuals).
Figure 12: Mobility tool ownership comparison with 2010

3.2 Chosen Mode

Most of the values for this analysis fall into the expected regions on the graph. However, for those individuals who have a public transportation subscription in 2015, the number of public transportation trips per day is greater than anticipated. This difference could be due to the definition of what counts as a season ticket. For the 2015 data, a GA card, Strecken-abo or verbund-abo counts as a season ticket, and a car is only considered in possession if it is always available. Figure 13 plots all of the collected data on trips compared to mobility tool ownership, and Figure 14 adds additional information on the number of non-motorised trips that are taken for each group. In Figure 14, the non-motorised transportation is displayed in the size of the bubble, and it consists of walking, biking and riding e-bikes. Due to the lack of information in previous micro-censuses, only data from 2005, 2010 and 2015 is included in the analysis.
Figure 13: Trips per day by mobility tool ownership
3.3 Daily Distance Travelled

While the graph is somewhat difficult to read, it is evident that the possession of a car allows for further distances to be travelled on average. In addition, those who have public transport subscriptions also travel further than their counterparts who do not have subscriptions. In order to follow the trends more clearly, separate graphs are added for each mobility tool ownership combination.

**Important Note:**

For the next set of graphs (where the mobility patterns are analysed by mobility tool possession), the original file received only contained data up to 2000. Thus, the data from the 2005, 2010 and 2015 micro-censuses is added in this iteration of the analysis.
Figure 15: Sum of daily distance travelled
3.4 Number of Trips per Day

Though each mobility tool ownership category is not distinctly separated, there is a correlation between car ownership and increased number of daily trips. For those who own a car, when adding a public transport subscription no clear differences may be seen. However, for those who do not have a car available, having a transportation subscription increases the number of trips.
Figure 16: Sum of daily distance travelled
3.5 **Sum of Daily Travel Time**

While distance and number of trips increased with the possession of a car, daily travel time increases with the ownership of a public transportation subscription. This follows as predicted since the same point to point trip usually takes longer by public transport than by car.
Figure 17: Sum of daily distance travelled
3.6 Average Travel Speed

Travel speed relates directly to the possession of public transport subscriptions or always available cars. With a car, the speeds are on average ten km/hr faster than travel by public transport. It is important to note that the public transport time also includes waiting time, which lowers the average speed.
Figure 18: Average travel speed by mobility tool
4 Analysis by Income Terciles

The income terciles in the analysis are manually calculated for each micro-census from 2000 onwards where the household income information is available. As income is a sensitive topic to many, there are a high proportion of system missing values. These have not been imputed for this analysis. The following graph displays the percentages of the population that fall into each tercile and the percentage accounted for by system missing. The population shares of the terciles cannot be precise due to the income categories of the respective micro-census.

In this analysis, the same methodology for calculating income terciles is applied to each micro-census. The method takes into account the provided household income and the household size. As the income information is not exact (it is estimated in income intervals in the micro-census), the midpoint value is assigned to each range. The midpoint values are then divided by the square root of the number of individuals in the household. Using the square root accounts for the lessening impact of an additional person on the household costs. The tercile bounds are set differently for each micro-census and are each chosen in an attempt to create three terciles with equal numbers of observations, and therefore the terciles approximate 33\% of the valid population each as closely as possible.

Since the methodology for calculating income terciles in previous years could not be reconstructed, the entirety of these graphs are recreated in this analysis. Due to this, the graphs differ from those created in the previous mobility pattern analysis. Whilst they still follow the same trends, the gaps between the terciles vary in size.

Figure 19: Average travel speed by mobility tool
The SPSS scripts for these analyses are also included in the same sets as the previous two sections.

4.1 License Ownership

Especially amongst older generations, the wealthier households have a higher percentage of license ownership.

Figure 20: License ownership by income tercile

![License ownership by income tercile](image)

4.2 Car Availability

As expected, the wealthiest third of society are more likely to have a car available all the time than the other terciles. The cross over and close lines of the second and third terciles could be due to the method of calculating the terciles.
4.3 Public Transportation Subscription

For individuals around 20 years old, income has a limited impact on public transport subscriptions. The youngest cohorts have the highest overall subscription rate which follows as predicted since these individuals do not usually have regular access to cars. Amongst the younger cohorts, public transportation membership is notably higher for the third tercile than the second. However, as the average cohort age increases, the terciles blur, and PT subscription ownership becomes less related to income.
4.4 Daily Distance Travelled

On average, households in the lower income tercile make fewer trips of shorter distances and shorter durations than the households in the second and third terciles. Similarly the wealthier households travel more, further and for longer. The trend between travel distance/time/frequency and income is potentially related to the increase in public transportation and car ownership by wealthier households and thereby the method of transportation that these households usually use.
Figure 23: Distance travelled by income tercile
4.5 Average Number of Trips

Figure 24: Number of trips by income tercile
4.6 Daily Travel Time

Figure 25: Travel time by income tercile

4.7 Average Travel Speed

The average travel speed of the third income tercile is notably quicker than the first and second terciles for most of the age cohorts. As the average age increases, the differences in speed between income terciles lessens as well. This indicates that after a certain age, income plays a smaller role for travel speed.
5 Introduction to Activity Chain Analysis

In the activity analyses, the average age of the cohort is determined in the same manner as in the mobility analysis. With a dearth of information about how the activity analysis was formerly conducted, this analysis is performed entirely separate from previous years and strives to loosely follow spirit of the earlier work. Only data from 2005, 2010 and 2015 is included because the previous micro censuses do not contain enough information to follow the same format of analysis as for the newer micro censuses.

For the purpose of clear analysis, the twelve to thirteen categories of activities provided in the micro censuses are merged into six unique groups. The table displays these allocations to the groups.
### Table 1: Activity Grouping

<table>
<thead>
<tr>
<th>WORK</th>
<th>LEISURE</th>
<th>SHOPPING</th>
<th>EDUCATION</th>
<th>BUSINESS</th>
<th>OTHER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work</td>
<td>Leisure</td>
<td>Shopping</td>
<td>Education</td>
<td>Business</td>
<td>Dienstfahrt</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Accompanying Children</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Accompanying Not Children</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Other</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Overnight</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Foreign Property</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Pseudoetappe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Change</td>
</tr>
</tbody>
</table>

The time spent at each activity is calculated from the micro census data using the departure and arrival times of each trip in conjunction with the trip purpose. All activities or journeys that begin after 23:59 on the specified day (or 1440 minutes past midnight in the data set) are excluded from this analysis as they disturb the calculation of hours spent at home. The hours at home as with the rest of the data is understood to occur in one twenty-four hour period.

In Appendices A.5, A.6 and A.7, the SPSS Syntax scripts for the activity chain analysis can be found.

### 5.1 Home and Out-of-Home Activities by Time

Time spent in out-of-home activities is highest in the late teens to early twenties, and gradually decreases as the population ages. Much of this decline may be explained by increased responsibility at home to spouses, children or even pets. As people age, they build stronger relationships with the others whom they share a home with, and they have increased responsibility of maintaining their home.
Figure 27: Time at home and out-of-home activities

Figure 28: Time at home and out-of-home activities by gender
5.2 Work Time

The plot of average time spent working for each age cohort matches the societal understanding about work whereby the working adult is usually between 20 and 65 years old. A sharp increase at either end of the graph portrays the uniformity with which individuals of a certain age join and leave the workforce. The plateau around 3.75 hours at work per day intuitively appears very low, however this value includes those who do not work, those who work part time and the respondents whose reporting day was on a day off work. Among the young and old age cohorts, the difference between men and women is only slight. However, for the majority of the working population women work significantly fewer hours than men do, on the range of one to one and a half hours.

Figure 29: Time spent at work
Figure 30: Time spent at work by gender

5.3 Business Activity Time

The time spent in business largely corresponds to the time spent at work graph. In this analysis, the trend is less consistent over the years, as there is a distinct increase in the average time spent on business related activities. The increase in business trips and business activities outside of the workplace indicates that individuals are more mobile now in their work and have wider connections of clients and co-workers. As with time spent at work, men occupy more of their time at business activites than women. This could be due to differences in the level of professional position between the genders, as men are more frequently employed in top-level positions which would require more business travel.
Figure 31: Time spent at business activities

Figure 32: Time spent at business activities by gender
5.4 Leisure Activity Time

The time spent on leisure peaks twice around the ages of 20 and 70 years. This may be explained by the responsibilities of individuals at those ages. Both age groups do not have many responsibilities to other people (namely families) aside from themselves or their significant other. When comparing the leisure graph to the working time graph, it is evident that increased working hours takes away time from activities such as leisure. Unlike the 30 to 60 year olds who have families and possibly high-level jobs, most twenty year olds are in education or new to the workforce and most seventy year olds are recently retired with grown up children or no children at all. On the far sides of both peaks, individuals lose mobility in childhood and old age, which limits their ability to take part in many leisure activities.

Figure 33: Time spent at leisure activities
5.5 Time in Education

The time spent in education is also indicative of the commonly understood structure of society. Individuals attend primary and secondary school until they are 16, and then many people attend additional schools through their early twenties. After reaching the age of thirty, few individuals remain in school and the graph displays this as time in education plateaus around 0 hours. There is no clear difference between genders.
Figure 35: Time spent in education

Figure 36: Time spent in education by gender
5.6 Shopping Time

Time spent shopping first plateaus at 30 years old, then increases after the age of 65 years. Most individuals from 20 to 65 years old have similar responsibilities and daily structures (work, family, etc), and thereby have similar shopping trends. Once individuals leave the workforce for retirement, they have more time to spend on activities such as shopping. Though the graph appears to vary greatly for different age groups it is important to note that the gap between 80 year olds and 30 year olds is less than five minutes per day spent shopping.

Figure 37: Time spent shopping
5.7 Time in Other Activities

The category of other activities includes accompanying children or disabled individuals, errands and uses of services, overnight visits and changes in transportation mode. This selection makes up for a very small percentage of the average individual’s time. Over the past ten years, the time spent in these other activities has increased but only marginally. The greatest difference between men and women occurs at the time of young families and may be explained by the tradition of women remaining at home to take care of children.
Figure 39: Time spent in other activities

![Figure 39](image)

Figure 40: Time spent in other activities by gender

![Figure 40](image)
5.8 Out-of-Home Time split by Activity

In the following graph, the combined trends of the previous graphs are clearly visible. In the case of shopping and other, the differences between age groups appear to be very minor. Even within leisure, the two peaks are less visible. What stands out the most is the transition from education to the workforce and then out of the work force. There is a clear trade-off between education and work for as time in education decreases the time at work increases at a similar rate.

Figure 41: Time spent in other activities

5.9 Number of Journeys

The data for number of journeys by each activity type comes from target person file. The shapes of the graphs for number of journeys per activity mirror those for time spent at each activity. On average, the highest number of journeys per day comes from leisure activities. With short activity time spans, leisure activities are completed more frequently and with greater variety. Due to this, more trips are required for leisure activities even though less time is spent on them than work or education.
5.10 Distance of Journeys for Activities

Due to a lack of historical data, the distance travelled to reach specific activities can only be calculated for the 2010 and 2015 datasets. Since only two years worth of data are available only allows a summary graph is displayed below. Unlike the other activity chain analyses, the distance measurement is an average of the trips to that activity instead of an average of the whole population.

Figure 43: Summary of average distance travelled for all activity types
6 Reference list


A SPSS Scripts

A.1 Mobility Analysis 2000

* Encoding: windows-1252.

*GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2000\spss\2000\zp.sav'.
*DATASET NAME cohort2000 WINDOW=FRONT.

*SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2000cohort.sav' /COMPRESSED.

GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2000cohort.sav'.
DATASET NAME cohort2000 WINDOW=FRONT.
weight by WP.

*remove unnecessary variables from the zielerson file.
delete variables f66100 f66300 f67000 f67010 f67020 f67030 f67101 airport1 f67200 f67210 f67220 f67300 f67400 f67410 f67420 f67601 zielort f67602 zstadt f67801 airport2 f69700 f70000 f70201 f70202 f70203 f70211 f70212a f70212b f70213 f70214a f70214b f70214c f70215 f70216a f70216b f70216c f70217 f70218a f70218b f70218c f70221 f70222 f70223 f70231 f70232 f70233 f70234 f70301 f70302 f70303 f70311 f70312 f70313 f70321 f70322 f70324 f70331 f70332 f70333 f70334 f70335 f70341 f70400 f70500 f50100 f50200 f50210 f50300 f50400 f50500 f50600 f50700 be ruf f51000 f51300 f51500 f51600 f51700 f52100 f52200 f52300 f52601 f52602 f52603 f60100 f60200 f60500 f60600 f60700 f64100 f64200 f65000 f65100 f65100 f65900 f66000 f50003 f50004 intdaur daysttag saison kanton.
execute.

*add the key variables from the household file.
STAR JOIN
/SELECT t0.person, t0.intnr, t0.zp, t0.wp, t0.f50001, t0.f50002, t0.f50005, t0.f50006, t0.f52604, t1.wm, t1.hhanzper, t1.ein, t1.f10700, t1.f30100, t1.income_tercile
/AS t1 /FROM * AS t0
/JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2000\spss\2000\hh.sav' AS t1 ON t0.hhnr=t1.hhnr
/OUTFILE FILE=''.

*add the key variables from the household person file.
STAR JOIN
/SELECT t0.intnr, t0.zp, t0.wp, t0.f50001, t0.f50002, t0.f50005, t0.f50006, t0.f52604, t0.wm, t0.hhanzper, t0.ein, t0.f10700, t0.f30100, t0.income_tercile, t1.f21000, t1.f21100, t1.f21200, t1.f22001, t1.f22002, t1.f22003
/AS t0
/JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2000\spss\2000\hp.sav' AS t1 ON t0.hhnr=t1.hhnr
AND t0.person = t1.person
/OUTFILE FILE=*.

*add the calculated trip time, distance and number of trips.

STAR JOIN
/SELECT t0.hhnr, t0.person, t0.zp, t0.wp, t0.f50001, t0.f50002, t0.f50005, t0.f50006, t0.f52604, t0.wm, t0.hhanzper, t0.ein, t0.f10700, t0.f30100, t0.income_tercile, t0.f21000, t0.f21100, t0.f21200, t0.f22001, t0.f22002, t0.f22003, t1.trip_num, t1.day_dist, t1.day_time
/FROM * AS t0
/JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2000trip,distance,time.sav' AS t1
ON t0.intnr = t1.INTNR
/OUTFILE FILE=*.

*add the total drivers license count per household.

*the dl_count was calculated in the same manner as for the 2015 data (with excel data taken from the hhper file).
*(countifs the DL variable = 1 and the HHNR equals the one in the receiving HHNR column).

STAR JOIN
/SELECT t0.intnr, t0.person, t0.zp, t0.wp, t0.f50001, t0.f50002, t0.f50005, t0.f50006, t0.f52604, t0.wm, t0.hhanzper, t0.ein, t0.f10700, t0.f30100, t0.income_tercile, t0.f21000, t0.f21100, t0.f21200, t0.f22001, t0.f22002, t0.f22003, t0.trip_num, t0.day_dist, t0.day_time, t1.DL_count
/FROM * AS t0
/JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2000dlcount.sav' AS t1
ON t0.hhnr = t1.hhnr
/OUTFILE FILE=*.

*------------------.
*Age Cohorts.
*------------------.
*coding to put people in brackets based on their age (in the 10 year brackets used for the Mobility data report).
*these age cohorts brackets differ from all other cohorts as they take the ages of 2000-YEAR instead of 2000-YEAR-1 which the other MZ analyses did.
recode f50001 (91 thru 100 = 0) (71 thru 90 = 1) (61 thru 70 = 2) (51 thru 60 = 3) (41 thru 50 = 4) (31 thru 40 = 5) (21 thru 30 = 6) (18 thru 20 = 7) (0 thru 17 = 9) into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
0 'before 1910'
1 '1910 - 1929'
2 '1930 - 1939'
3 '1940 - 1949'
4 '1950 - 1959'
5 '1960 - 1969'
6 '1970 - 1979'
7 '1980 - 1989'
9 'under 18 years old and not used for analysis'.
formats age_cohort(f8.0).

*THIS AGE COHORT AVERAGE COMES FROM THE WEIGHTED AVERAGE.
weight WP.
*finding the average age for each cohort.
means f50001 by age_cohort
   /cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (0=92.72) (1 = 77.58) (2=65.46) (3=55.48) (4=45.54) (5=35.62) (6=25.76) (7=19.01) (9=11.60) into cohort_average_age.
variable labels cohort_average_age 'The average age of all of the individuals in the cohort'.
value labels cohort_average_age
92.72 'before 1910'
77.58 '1910 - 1929'
65.46 '1930 - 1939'
55.48 '1940 - 1949'
45.54 '1950 - 1959'
35.62 '1960 - 1969'
25.76 '1970 - 1979'
19.01 '1980 - 1989'.
formats cohort_average_age(f8.2).
execute.

*delete the data points where the age is below 18.
SELECT IF age_cohort ne 9.
execute.

*------------------.
*Neatening the raw variables for use.
*------------------.
*change the labels to English labels.
variable labels
hhnr 'household number'
person 'person number'
zp 'zielperson number'
wp 'person weight'
f50001 'age'
f50002 'gender'
f50005 'drivers license'
f50006 'motorcycle license'
f52604 'car availability'
w m 'household weight'
hhanzper 'number of people in the household'
ein 'monthly household income'
f10700 'type of household'
f30100 'number of cars in the household'
f21000 'possession of Halbtax'
f21100 'other pt subscription'
f21200 'GA pt'
f22001 'number of weekly subscriptions'
f22002 'number of monthly subscriptions'
f22003 'number of annual subscriptions'
trip_num 'number of daily trips'
day_dist 'total daily distance from trips (km)'
day_time 'total travel time in daily trips (min)'
DL_count 'number of drivers licenses in the household'.
execute.

*remove negatives.
recode all (-1 = sysmis) (-2 = sysmis).
*recode no's to 0.
recode f50005 f50006 (2 = 0).
*recode the car availability to always available.
recode f52604 (2 thru 3 = 0).
do if f50005 eq 0.
   recode f52604 (sysmis =0).
end if.
execute.

*update the value labels in english.
value labels f50005 1 'yes' 0 'no' /
f50006 1 'yes' 0 'no' /
f52604 1 'always available' 0 'not always available' /
f50002 1 'male' 2 'female'.
execute.

*------------------.
*Calculate the pt_sub pt_sub_1 and pt_all variables.
*------------------.
*pt_sub is just (GA, strecke and inter/verbund).
*pt_sub_1 includes all public transportation subscriptions (GA, strecke, inter and other).
*pt_all includes anything related to a public transport any variable with oevabo (GA....Halbtax).
*pt_all is the value that was used in Professor Axhausen's analysis of the mobility patterns in Switzerland even though he agrees it shouldn't be this way.
*pt_sub.

if f21200 = 1 pt_sub = 1.
recode f22001 (1 thru 3 = 1) into pt_sub.
recode f22002 (1 thru 3 = 1) into pt_sub.
recode f22003 (1 thru 3 = 1) into pt_sub.
recode pt_sub (sysmis = 0).
variable labels pt_sub 'Public transportation subscription (GA, verbund, strecke)'.
formats pt_sub (f8.0).
execute.
*pt_sub_1.
if f21200 = 1 pt_sub_1 = 1.
recode f22001 (1 thru 3 = 1) into pt_sub_1.
recode f22002 (1 thru 3 = 1) into pt_sub_1.
recode f22003 (1 thru 3 = 1) into pt_sub_1.
if f21100 = 1 pt_sub_1 = 1.
recode pt_sub_1 (sysmis = 0).
variable labels pt_sub_1 'Public transportation subscription including other subscriptions in addition to the pt_sub values'.
formats pt_sub_1 (f8.0).
execute.

*pt_all.
if f21000 = 1 pt_all = 1.
if f21200 = 1 pt_all = 1.
recode f22001 (1 thru 3 = 1) into pt_all.
recode f22002 (1 thru 3 = 1) into pt_all.
recode f22003 (1 thru 3 = 1) into pt_all.
if f21100 = 1 pt_all = 1.
recode pt_all (sysmis = 0).
variable labels pt_all 'Possession of any public transport subscription or discount card'.
formats pt_all (f8.0).
execute.

*------------------.
*Car per drivers license in the household.
*------------------.
do if sysmis(f30100).
    compute car_per_license= -9.
recode car_per_license (-9 = sysmis).
else if f30100 = 0.
    compute car_per_license = 0.
else if DL_count = 0.
    compute car_per_license = 0.
ELSE.
    compute car_per_license = f30100 / DL_count.
end if.
variable labels car_per_license 'Ratio of cars to drivers licenses in the household'.
execute.

*------------------.
*Classifications for with and without PT sub and passenger car available.
*------------------.
*using the pt_sub that doesn't include the discount cards like the Halbtax.
if pt_sub eq 1 and f52604 eq 1 pt_pw eq 1.
if pt_sub eq 0 and f52604 eq 1 pt_pw eq 2.
if pt_sub eq 1 and f52604 eq 0 pt_pw eq 3.
if pt_sub eq 0 and f52604 eq 0 pt_pw eq 4.
variable labels pt_pw 'Possession of public transport subscriptions and/or passenger vehicles'.
value labels pt_pw
   1 'PT subscription and car available'
   2 'no PT subscription but car available'
   3 'PT subscription but no car available'
   4 'neither of either'.
formats pt_pw (f8.0).
frequencies pt_pw.
execute.

*------------------.*
*INCOME TERCILES.*
*------------------.*
*Creating a variable for the lower and upper bounds of the income bracket that is provided in the original data.*
recode ein (1 = 1000) (2 = 3000) (3 = 5000) (4 = 7000) (5 = 9000) (6 = 11000) (7 = 13000) (8 = 15000) (9 = 17000) into hausalts_einkommen_mid.
compute eff_income = hausalts_einkommen_mid / sqrt(hhanzper).
execute.

*display frequencies to verify.
FREQUENCIES eff_income.

*Setting the income terciles accounting for the number of people per household.
do if missing(eff_income).
else.
   compute income_tercile = 2.
   recode eff_income (5000 THRU HIGHEST=3) into income_tercile.
   recode eff_income (LOWEST THRU 3480=1) into income_tercile.
end if.
execute.

*English variable labels.
VARIABLE LABELS income_tercile 'The income terciles'.
add value labels income_tercile 1 'First tercile' 2 'Second tercile' 3 'Third tercile'.
formats income_tercile (f2.0).
VARIABLE WIDTH income_tercile (8).
execute.

*display frequencies to verify.
frequencies income_tercile.
execute.

*------------------.*
*Income Tercile by Gender.*
*------------------.*
*to find the percentage of each tercile that is made up of each gender, and to see if there is a difference between genders with respect to income.

do if f50002 = 1.
    compute income_tercile_gender = income_tercile.
else if f50002 = 2.
    recode income_tercile (1 = 4) (2 = 5) (3 = 6) into income_tercile_gender.
end if.
execute.

variable labels income_tercile_gender 'Income terciles separated by gender'.
value labels income_tercile_gender
1 'Male: first tercile'
2 'Male: second tercile'
3 'Male: third tercile'
4 'Female: first tercile'
5 'Female: second tercile'
6 'Female: third tercile'.

formats income_tercile_gender (f8.0).
frequencies income_tercile_gender.
execute.

*------------------.
*Average travel speed.
*------------------.
do if day_time gt 0.
    compute speed = day_dist / day_time *60.
else if day_time eq 0.
    compute speed = -99.
    recode speed (-99 = sysmis).
end if.

variable labels speed 'Daily travel speed in km / hr'.
execute.

do if f52604 = 1.
    compute car_speed = speed.
end if.

variable labels car_speed 'Travel speed only for those with a car always available'.
execute.

means car_speed by age_cohort by f50002
   /cells mean stddev count min max.

T-TEST GROUPS=f50002(1 2)
   /MISSING=ANALYSIS
   /VARIABLES=car_speed
   /CRITERIA=CI(.95).

*------------------.
*Collect Weighted Average Data.
*------------------.
*Display the data that is then collected in Excel for graphic construction.
weight by WP.
execute.

*Average car per license by age and gender.
means car_per_license by age_cohort by f50002
   /cells = mean stddev count sum.

*Average car availability by age and gender.
means f52604 by age_cohort by f50002
   /cells = mean stddev count sum.

*PT OWNERSHIP AND CAR AVAILABILITY SPLIT.
*Average daily travel distance (domestic) by age and ownership.
means day_dist by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and ownership.
means trip_num by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trip time (domestic) by age and ownership.
means day_time by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily travel speed (domestic) by age and ownership.
means speed by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and gender.
means trip_num by age_cohort by f50002
   /cells = mean stddev count min max.

*Average daily travel speed (domestic) by age and gender.
means speed by age_cohort by f50002
   /cells = mean stddev count min max.

*Average public transportation subscription by age and gender.
means pt_sub by age_cohort by f50002
   /cells = mean stddev count.

*INCOME TERCILE DATA.
*Average drivers license ownership by age and income.
means f50005 by age_cohort by income_tercile
   /cells = mean stddev count.
*Average car availability by age and income.
means f52604 by age_cohort by income_tercile
   /cells = mean stddev count.

*Average public transportation subscription by age and income.
means pt_sub by age_cohort by income_tercile
   /cells = mean stddev count.

*Average daily travel distance (domestic) by age and income.
means day_dist by age_cohort by income_tercile
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and income.
means trip_num by age_cohort by income_tercile
   /cells = mean stddev count min max.

*Average daily trip time (domestic) by age and income.
means day_time by age_cohort by income_tercile
   /cells = mean stddev count min max.
execute.

*Average daily travel speed (domestic) by age and income.
means speed by age_cohort by income_tercile
   /cells = mean stddev count min max.

*------------------.
*Mobility tool ownership split by gender.
*------------------.
do if f50002 = 1.
   recode pt_pw (1=1) (2=2) (3=3) (4=4) into pt_pw_gesl.
else if f50002 = 2.
   recode pt_pw (1=5) (2=6) (3=7) (4=8) into pt_pw_gesl.
end if.
variable labels pt_pw_gesl 'Mobility tool ownership split by gender'.
value labels pt_pw_gesl
  1 'PT and car - male'
  2 'car and no PT - male'
  3 'PT and no car - male'
  4 'no PT no car - male'
  5 'PT and car - female'
  6 'car and no PT - female'
  7 'PT and no car - female'
  8 'no PT no car - female'.
execute.
frequencies pt_pw_gesl.
A.2 Mobility Analysis 2005

* Encoding: windows-1252.
*GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2005\MZ05_Datenbank_CH\2_DB_SPSS\Zielpersonen.sav'.
*DATASET NAME cohort2005 WINDOW=FRONT.

*SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2005cohort.sav' /COMPRESSED.

GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2005cohort.sav'.
DATASET NAME cohort2005 WINDOW=FRONT.

weight by WP.

*remove the unnecessary variables from the zielpersonen file.

STAR JOIN
/SELECT t0.F22A, t0.T_WEGE, t0.ZIELPNR, t0.WP, t0.F415E, t0.T_ETAPPEN, t0.T_DIST, t0.T_WEGZEIT,
t0.F23B, t0.F23C, t0.F24, t0.F25, t0.F26, t1.F01, t1.F21, t1.F31
/ FROM * AS t0
JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past '+
'Microcensuses\MZ2005\MZ05_Datenbank_CH\2_DB_SPSS\Haushalte.sav' AS t1
ON t0.HHNR=t1.HHNR
OUTFILE FILE=*.

*add the total drivers license count per household.
the dl_count was calculated in the same manner as for the 2015 data (with excel data taken from the hhper file).

*(countifs the DL variable = 1 and the HHNR equals the one in the receiving HHNR column).

STAR JOIN
/SELECT t0.F22A, t0.T_WEGE, t0.ZIELPNR, t0.WP, t0.F415E, t0.T_ETAPPEN, t0.T_DIST, t0.T_WEGZEIT,
t0.F23B, t0.F23C, t0.F24, t0.F25, t0.F26, t1.F01, t1.F21, t1.F31
/ FROM * AS t0
JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2005dlcount.sav' AS t1
ON t0.HHNR=t1.HHNR
OUTFILE FILE='.'. 

*... Neatening the raw variables for use.
*variable labels
F415E 'car availability'
F26 'Has one or more public transportation subscription'
F23B 'age'
F23C 'sex'
F25 'motorcycle license'
T_DIST 'daily travel distance'
T_WEGE 'number of trips'
T_WEGZEIT 'daily trip time in min domestic total'
F22A 'Number of people in household'
F01 'Household income'
F21 'type of household'
f31 'number of cars in household'
dl_count 'number of licenses in household'.

*recode the variables so there are no negative values.
recode all (-99 = sysmis) (-98 = sysmis) (-97 = sysmis).
execute.
*recode the license records so 0 is no.
recode F24 (2 = 0).
recode F25 (2 = 0).
value labels F24 1 'Yes' 0 'No'.
value labels F25 1 'Yes' 0 'No'.
execute.

*recode car availability for always available.
recode F41E (2 thru 4 =0).
do if f24 eq 0.
  recode f41E (sysmis = 0).
end if.
value labels F41E 1 'always available' 0 'not always available'.

*-------------------------------------------------- ----------------------------------.
*Age Cohorts.
*-------------------------------------------------- ----------------------------------.
*coding to put people in brackets based on their age (in the 10 year brackets used for the Mobility data report).
*The age cohorts are calculated as the census date minus the bracket limits minus 1.
*The extra year subtracted accounts for the fact that the census was taken at the start of the year mentioned (I think it complicates it unnecessarily but hey).
recode F23B (95 thru 100 = 0) (75 thru 94 = 1) (65 thru 74 = 2) (55 thru 64 = 3) (45 thru 54 = 4) (35 thru 44 = 5) (25 thru 34 = 6) (18 thru 24 = 7) (0 thru 17 = 9) into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
 0 'before 1910'
 1 '1910 - 1929'
 2 '1930 - 1939'
 3 '1940 - 1949'
 4 '1950 - 1959'
 5 '1960 - 1969'
 6 '1970 - 1979'
 7 '1980 - 1989'
 9 'under 18 years old and not used for analysis'.
formats age_cohort(f8.0).

*AVERAGE AGE USES THE PERSON WEIGHTS.
weight by WP.
*finding the average age for each cohort.
means F23B by age_cohort
  /cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (0 = 96.42) (1 = 80.11) (2=69.29) (3=59.28) (4=49.23) (5=39.57) (6=29.75) (7=20.82) into cohort_average_age.
variable labels cohort_average_age 'The average age of all of the individuals in the cohort'.
value labels cohort_average_age
 80.11 '1910 - 1929'
 69.29 '1930 - 1939'
 59.28 '1940 - 1949'
 49.23 '1950 - 1959'
 39.57 '1960 - 1969'
 29.75 '1970 - 1979'
 20.82 '1980 - 1989'.
formats cohort_average_age(f8.2).
execute.

*delete the data points where the age is below 18 and the age is above 94 (there aren't enough data points to validate the before 1910 cohort).
SELECT IF age_cohort ne 9 and age_cohort ne 0.
execute.

*Calculate the pt_sub pt_sub_1 and pt_all variables.
*-------------------------------------------------- ----------------------------------.
*pt_sub is just (GA, strecke and inter/verbund).
*pt_sub_1 includes all public transportation subscriptions (GA, strecke, inter and other).
*pt_all includes anything related to a public transport any variable with oevabo (GA,...Halbtax).
*pt_all is the value that was used in Professor Axhausen's analysis of the mobility patterns in Switserland even though he agrees it shouldn't be this way.

*pt_sub.
recode F26 (2 thru 7 = 1) (12 thru 15 = 1) into pt_sub.
recode pt_sub (sysmis = 0).
variable labels pt_sub 'Public transportation subscription (GA, verbund, strecke)'.
formats pt_sub (f8.0).
execute.

*pt_sub_1.
recode F26 (2 thru 7 = 1) (12 thru 15 = 1) (10 = 1) (17 = 1) into pt_sub_1.
recode pt_sub_1 (sysmis = 0).
variable labels pt_sub_1 'Public transportation subscription including other subscriptions in addition to the pt_sub values'.
formats pt_sub_1 (f8.0).
execute.

*pt_all.
recode F26 (1 thru 10 = 1) (12 thru 20 = 1) into pt_all.
recode pt_all (sysmis = 0).
variable labels pt_all 'Possession of any public transport subscription or discount card'.
formats pt_all (f8.0).
execute.

*-------------------------------------------------- ----------------------------------.
*Classifications for with an without PT sub and passenger car available.
*------------------------------------------------------------------------------------------------------------------.
*using the pt_sub that doesn’t include the discount cards like the Halbtax.
if pt_sub eq 1 and F415E eq 1 pt_pw eq 1.
if pt_sub eq 0 and F415E eq 1 pt_pw eq 2.
if pt_sub eq 1 and F415E eq 0 pt_pw eq 3.
if pt_sub eq 0 and F415E eq 0 pt_pw eq 4.
variable labels pt_pw 'Possession of public transport subscriptions and/or passenger vehicles'.
value labels pt_pw
 1 'PT subscription and car available'
 2 'no PT subscription but car available'
 3 'PT subscription but no car available'
 4 'neither of either'.
formats pt_pw (f8.0).
frequencies pt_pw.
execute.

*------------------------------------------------------------------------------------------------------------------.
*INCOME TERCILES.
*------------------------------------------------------------------------------------------------------------------.
*Creating a variable for the lower and upper bounds of the income bracket that is provided in the original data.
recode F101 (1 = 1000) (2 = 3000) (3 = 5000) (4 = 7000) (5 = 9000) (6 = 11000) (7 = 13000) (8 = 15000) (9 = 17000) into hausalts_einkommen_mid.
compute eff_income = hausalts_einkommen_mid / sqrt( F22A).
execute.

*display frequencies to verify.
FREQUENCIES eff_income.

*Setting the income terciles accounting for the number of people per household.
do if missing(eff_income).
else.
  compute income_tercile = 2.
  recode eff_income (5000 THRU HIGHEST=3) into income_tercile.
  recode eff_income (LOWEST THRU 3480=1) into income_tercile.
end if.
execute.

*English variable labels.
VARIABLE LABELS income_tercile 'The income terciles'.
add value labels income_tercile 1 'First tercile' 2 'Second tercile' 3 'Third tercile'.
formats income_tercile (f2.0).
VARIABLE WIDTH income_tercile (8).
execute.

*display frequencies to verify.
frequencies income_tercile.
execute.

*-------------------------------------------------- ----------------------------------.
*Income Tercile by Gender.
*------------------------------------------------------------------------------------------------------------------.
*to find the percentage of each tercile that is made up of each gender, and to see if there is a difference between genders with respect to income.
do if f23c = 1.
   compute income_tercile_gender = income_tercile.
else if f23c = 2.
   recode income_tercile (1 = 4) (2 = 5) (3 = 6) into income_tercile_gender.
end if.
execute.
variable labels income_tercile_gender 'Income terciles separated by gender'.
value labels income_tercile_gender
   1 'Male: first tercile'
   2 'Male: second tercile'
   3 'Male: third tercile'
   4 'Female: first tercile'
   5 'Female: second tercile'
   6 'Female: third tercile'.
formats income_tercile_gender (f8.0).
frequencies income_tercile_gender.
execute.
*------------------------------------------------------------------------------------.
*Car per drivers license weight with number of cars.
*------------------------------------------------------------------------------------.
do if f415e = 1.
do if sysmis(f31).
   compute car_per_license = -9.
else if f31 = 0.
   compute car_per_license = 0.
else if dl_count = 0.
   compute car_per_license = 0.
ELSE.
   compute car_per_license = f31 / dl_count.
end if.
*end if.
variable labels car_per_license 'Ratio of cars to drivers licenses in the household'.
formats car_per_license (f8.2).
if car_per_license gt 1 car_per_license eq 1.
execute.
*------------------------------------------------------------------------------------.
*Average travel speed.
*------------------------------------------------------------------------------------.
do if T_WEGZEIT gt 0.
   compute speed = T_DIST / T_WEGZEIT * 60.
else if T_WEGZEIT eq 0.
   compute speed = -99.
   recode speed (-99 = sysmis).
end if.
variable labels speed 'Daily travel speed in km / hr'.
execute.
do if F415E = 1.
   compute car_speed = speed.
end if.
variable labels car_speed 'Travel speed only for those with a car always available'.
execute.
means car_speed by age_cohort by f23c
   /cells mean stddev count min max.
*------------------------------------------------------------------------------------.
*Number of bike/walk trips.
*------------------------------------------------------------------------------------.
GET FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2005\MZ05_Datenbank_CH\CH2_DB_SPSSWege.sav'.
dataset name walk_bike2005 window=front.
recode wmittel (14 thru 15 = 1) (else = 0) into walk_bike.
execute.
weight off.

DATASET DECLARE sum_walk_bike.
AGGREGATE /OUTFILE='sum_walk_bike'
/BREAK=HHNR ZIELPNR
/walk_bike_trips=SUM(walk_bike).

dataset activate sum_walk_bike.
variable labels walk_bike_trips 'Number of trips taken by foot or by bike'.

dataset activate cohort2005.
STAR JOIN
/SELECT t0.F22A, t0.T_WEGE, t0.WP, t0.F415E, t0.T_ETAPPEN, t0.T_DIST, t0.T_WEGZEIT, t0.F23B,
t0.F23C, t0.F24, t0.F25, t0.F26, t0.F101, t0.F21, t0.F31, t0.DL_count, t0.age_cohort,
t0.cohort_average_age, t0.pt_sub, t0.pt_sub_1, t0.pt_all, t0.pt_pw, t0.hausaus_einkommen_mid,
t0.eff_income, t0.income_tercile, t0.income_tercile_gender, t0.car_per_license, t0.speed,
t0.car_speed, t1.walk_bike_trips
/FROM * AS t0
/JOIN 'sum_walk_bike' AS t1
ON t0.HHNR=t1.HHNR
AND t0.ZIELPNR=t1.ZIELPNR
/OUTFILE FILE=''.

*Collect Weighted Average Data.
*-------------------------------------------------- ----------------------------------.
weight by WP.
execute.

*Average car per license by age and gender.
means car_per_license by age_cohort by f23c
   /cells = mean stddev count sum.

*Average car availability by age and gender.
means f415e by age_cohort by f23c
   /cells = mean stddev count sum.

*PT OWNERSHIP AND CAR AVAILABILITY SPLIT.

*Average daily travel distance (domestic) by age and ownership.
means T_DIST by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and ownership.
means T_WEGE by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trip time (domestic) by age and ownership.
means T_WEGZEIT by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily travel speed (domestic) by age and ownership.
means speed by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and gender.
means T_WEGE by age_cohort by F23C
   /cells = mean stddev count min max.

*Average daily travel speed (domestic) by age and gender.
means speed by age_cohort by F23C
   /cells = mean stddev count min max.

*Average public transportation subscription by age and gender.
means pt_sub_2 by age_cohort by F23C
   /cells = mean stddev count.

*INCOME TERCILE DATA.

*Average drivers license ownership by age and income.
means F24 by age_cohort by income_tercile
   /cells = mean stddev count.

*Average car availability by age and income.
*Average public transportation subscription by age and income.
means pt_sub by age_cohort by income_tercile
/cells = mean stddev count.

*Average daily travel distance (domestic) by age and income.
means T_DIST by age_cohort by income_tercile
/cells = mean stddev count min max.

*Average daily trips (domestic) by age and income.
means T_WEG by age_cohort by income_tercile
/cells = mean stddev count min max.

*Average daily trip time (domestic) by age and income.
means T_WEGZEIT by age_cohort by income_tercile
/cells = mean stddev count min max.

*Average daily travel speed (domestic) by age and income.
means speed by age_cohort by income_tercile
/cells = mean stddev count min max.

*Walk and bike trips by mobility tool ownership.
means walk_bike_trips by pt_pw
/cells = mean stddev count min max.

*Mobility tool ownership split by gender.
do if F23C = 1.
   recode pt_pw (1=1) (2=2) (3=3) (4=4) into pt_pw_gesl.
else if F23C = 2.
   recode pt_pw (1=5) (2=6) (3=7) (4=8) into pt_pw_gesl.
end if.
variable labels pt_pw_gesl 'Mobility tool ownership split by gender'.
value labels pt_pw_gesl
  1 'PT and car - male'
  2 'car and no PT - male'
  3 'PT and no car - male'
  4 'no PT no car - male'
  5 'PT and car - female'
  6 'car and no PT - female'
  7 'PT and no car - female'
  8 'no PT no car - female'.
execute.
frequencies pt_pw_gesl.

A.3 Mobility Analysis 2010

* Encoding: windows-1252.
*GET
  FILE=P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past
  Microcensuses\MZ2010\3_DB_SPSS\zielpersonen.sav'.
*DATASET NAME ziel2010 WINDOW=FRONT.

*SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010cohort.sav'
  /COMPRESSED.

*The final file.
GET
  FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010cohort.sav'.
DATASET NAME cohort2010 WINDOW=FRONT.
*remove the unnecessary variables from the zielpersonen file. delete variables proxi sprache dmd zeit_zp BTag US Tag VSTag tag f40200_01 f40202 f40300 f40400_01a f40400_01b f40400_01c f40400_01d f40400_01e f40402a f40402b f40402c f40402d f40402e f40500 f40600 f40700 f40701 f40702 f40800_01 f40900 f40901_02 f40903 f40904_05 f40906 f41000a f41000b f41000c f41001a f41001b f41001c f41100_01 f41102 f41100 f411300_01 f41400_01 f41500_01 f41700 f41800 f41900 f42000 f42100a f42100b f42100c f42100d f42100e f42100f f42100g f42100h f80500 f80600a f80600b f80600c f80600d Berufstamm f81200 f81300 f81400 f81500a f81500b f81500c f81500d f81500e f81500f f81500g f81500h f81500i f81500j A_X A_Y A_X_C1903 A_Y_C1903 A_QAL A_BFS A_PLZ A_Ort A_str A_strf A SPRACHE A_REGION A_KANTON A_LND A_AGGLO A_AGGLO_GROESSE A_STRUKTUR A_STRUKTUR_AGG A_struktur_bfs9 AU_X AU_Y AU_X_C1903 AU_Y_C1903 AU_QAL AU_BFS AU_PLZ AU_Ort AU_Ort AU_hnr AU_hnr_AU_SPRACHE AU_REGION AU_KANTON AU_LND AU_AGGLO AU_AGGLO_GROESSE AU_STRUKTUR AU_STRUKTUR_AGG AU_struktur_bfs9 W_X W_Y W_X_C1903 W_Y_C1903 W_QAL W_BFS W_SPRACHE W_REGION W_KANTON W_AGGLO W_AGGLO_GROESSE W_STRUKTUR W_STRUKTUR_AGG w_struktur_bfs9 AMSTAT ESTATUS BSTELL BSGRAD HAUSB ISCED noga_08 wiaht_08 SNB2000_1 ISCO_08 soc_cat soc_cat_II zivil nation f42299 f51500_sum f51500_sum_Inland rdist_sum e_dauer_sum T_ET APPEN T DISTLV T DISTMIV T DISTOEV T DISTAND T ARBEIT T USBILDUNG T EINKAUF T NUTZ T FREIZEIT T BEGLEIT.

*add income related variables from the household file (as well as the total number of vehicles). STAR JOIN

/SELECT t0.ZIELPNR, t0.WP, t0.alter, t0.gesl, t0.f20400a, t0.f20400b, t0.rdist_sum_Inland, t0.e_dauer_sum_Inland, t0.T_WEGE, t0.T_WEGZEIT, t1.hhtyp, t1.F20601, t1.hhgr, t1.income_lowerbound, t1.income_upperbound, t1.income_pp_low, t1.income_pp_high, t1.income_tercile /FROM * AS t0 /JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past '+ 'Microcensuses\MZ2010\3_DB_SPSS\haushalte.sav' AS t1 ON t0.HHNR=t1.HHNR /OUTFILE FILE=*.

*neatening the raw variables for use.

*change the labels to English labels. variable labels f41600a PT: 'half fare' f41600b PT: 'GA first class' f41600c PT: 'GA second class' f41600d PT: 'verbund' f41600e PT: 'route subscription' f41600f PT: 'track 7' f41600g PT: 'other' f42100e 'car availability' alter 'age' gesl 'sex' f20400a 'PW license' f20400b 'motorcycle license' rdist_sum_Inland 'daily travel distance' e_dauer_sum_Inland 'daily domestic travel time' T_WEGE 'number of trips' T_WEGZEIT 'daily trip time in min domestic total' f20601 'Household income' hhgr 'number of people in the household' hhtyp 'type of household'.
execute.

*recode the variables so they make more sense to work with (take out the 2=nein).
recode all (-99 = sysmis) (-98= sysmis) (-97 = sysmis).
recode f20400a f20400b (2=0).
execute.

*recode car availability for always available.
recode f42100e (2 thru 3 =0).
do if f20400a eq 0.
   recode f42100e (sysmis = 0).
end if.
execute.

value labels f42100e 1 'always available' 0 'not always available' /
f20400a 1 'yes' 0 'no' /
f20400b 1 'yes' 0 'no'.
execute.

*------------------------------------------------------------------------------------.
*Age Cohorts.
*-------------------------------------------------- ----------------------------------.
*coding to put people in brackets based on their age (in the 10 year brackets that Axhausen used for the Mobility data report).
*The age cohorts are calculated as the census date minus the bracket limits minus 1.
*The extra year subtracted accounts for the fact that the census was taken at the start of the year mentioned (I think it complicates it unnecessarily but hey).
recode alter (80 thru 100 = 1) (70 thru 79 = 2) (60 thru 69 = 3) (50 thru 59 = 4) (40 thru 49 = 5) (30 thru 39 = 6) (20 thru 29 = 7) (18 thru 19 = 8) (0 thru 17 = 9) into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
1 '1910 - 1929'
2 '1930 - 1939'
3 '1940 - 1949'
4 '1950 - 1959'
5 '1960 - 1969'
6 '1970 - 1979'
7 '1980 - 1989'
8 '1990 - 1999'
9 'under 18 years old and not used for analysis'.
formats age_cohort(f8.0).

*AVERAGE AGE USES THE PERSON WEIGHTS.
weight by WP.
*finding the average age for each cohort.
means alter by age_cohort
   /cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (1 = 84.11) (2=74.16) (3=64.26) (4=54.29) (5=44.41) (6=34.70) (7=24.59) (8=18.49) into cohort_average_age.
variable labels cohort_average_age 'The average age of all of the individuals in the cohort'.
value labels cohort_average_age
84.11 '1910 - 1929'
74.16 '1930 - 1939'
64.26 '1940 - 1949'
54.29 '1950 - 1959'
44.41 '1960 - 1969'
34.70 '1970 - 1979'
24.59 '1980 - 1989'
18.49 '1990 - 1999'.

*delete the data points where the age is below 18.

SELECT IF age_cohort ne 9.

*Calculate the pt_sub pt_sub_1 and pt_all variables.

*pt_sub_1 includes all public transportation subscriptions (GA, strecke, inter and other)
*pt_sub is just (GA, strecke and inter/verbund).
*pt_all includes anything related to a public transport any variable with oevabo (GA...Halbtax).
*pt_all is the value that was used in Professor Axhausen's analysis of the mobility patterns in Switerland even though he agrees it shouldn't be this way.

*pt_sub.
if f41600b = 1 pt_sub = 1.
if f41600c = 1 pt_sub = 1.
if f41600d = 1 pt_sub = 1.
if f41600e = 1 pt_sub = 1.
recode pt_sub (sysmis = 0).

variable labels pt_sub 'Public transportation subscription (GA, verbund, strecke)'.
formats pt_sub (f8.0).

*pt_sub_1.
if f41600b = 1 pt_sub_1 = 1.
if f41600c = 1 pt_sub_1 = 1.
if f41600d = 1 pt_sub_1 = 1.
if f41600e = 1 pt_sub_1 = 1.
if f41600g = 1 pt_sub_1 = 1.
recode pt_sub_1 (sysmis = 0).

variable labels pt_sub_1 'Public transportation subscription including other subscriptions in addition to the pt_sub values'.
formats pt_sub_1 (f8.0).

*pt_all.
if f41600a = 1 pt_all = 1.
if f41600b = 1 pt_all = 1.
if f41600c = 1 pt_all = 1.
if f41600d = 1 pt_all = 1.
if f41600e = 1 pt_all = 1.
if f41600f = 1 pt_all = 1.
if f41600g = 1 pt_all = 1.
recode pt_all (sysmis = 0).

variable labels pt_all 'Possession of any public transport subscription or discount card'.
formats pt_all (f8.0).

*Classifications for with and without PT sub and passenger car available.
*using the pt_sub that doesn't include the discount cards like the Halbtax.
if pt_sub eq 1 and f42100e eq 1 pt_pw eq 1.
if pt_sub eq 0 and f42100e eq 1 pt_pw eq 2.
if pt_sub eq 1 and f42100e eq 0 pt_pw eq 3.
if pt_sub eq 0 and f42100e eq 0 pt_pw eq 4.
variable labels pt_pw 'Possession of public transport subscriptions and/or passenger vehicles'.
value labels pt_pw
  1 'PT subscription and car available'
  2 'no PT subscription but car available'
  3 'PT subscription but no car available'
  4 'neither of either'.
formats pt_pw (f8.0).
frequencies pt_pw.
execute.

*----------------------------------------------------------------------------------------------.

*INCOME TERCILES.
*----------------------------------------------------------------------------------------------.

*Creating a variable for the lower and upper bounds of the income bracket that is provided in the original data.
recode F20601 (1 = 1000) (2 = 3000) (3 = 5000) (4 = 7000) (5 = 9000) (6 = 11000) (7 = 13000) (8 = 15000) (9 = 17000) into hausalts_einkommen_mid.
compute eff_income = hausalts_einkommen_mid / sqrt(hhgr).
execute.

*display frequencies to verify.
FREQUENCIES eff_income.

*Setting the income terciles accounting for the number of people per household.
do if missing(eff_income).
  else.
    compute income_tercile = 2.
    recode eff_income (5200 THRU HIGHEST=3) into income_tercile.
    recode eff_income (LOWEST THRU 3500=1) into income_tercile.
  end if.
execute.

*English variable labels.
VARIABLE LABELS income_tercile 'The income terciles'.
add value labels income_tercile 1 'First tercile' 2 'Second tercile' 3 'Third tercile'.
formats income_tercile (f2.0).
VARIABLE WIDTH income_tercile (8).
execute.

*display frequencies to verify.
frequencies income_tercile.
execute.

*----------------------------------------------------------------------------------------------.

*Income Tercile by Gender.
*----------------------------------------------------------------------------------------------.

*to find the percentage of each tercile that is made up of each gender, and to see if there is a difference between genders with respect to income.
do if gesl = 1.
  compute income_tercile_gender = income_tercile.
else if gesl = 2.
  recode income_tercile (1 = 4) (2 = 5) (3 = 6) into income_tercile_gender.
execute.
end if.
execute.

variable labels income_tercile_gender 'Income terciles separated by gender'.

value labels income_tercile_gender
1 'Male: first tercile'
2 'Male: second tercile'
3 'Male: third tercile'
4 'Female: first tercile'
5 'Female: second tercile'
6 'Female: third tercile'.

formats income_tercile_gender (f8.0).

frequencies income_tercile_gender.
execute.

*------------------------------------------------------------------------------------.
*Average travel speed.
*------------------------------------------------------------------------------------.

do if T_WEGZEIT gt 0.
   compute speed = rdist_sum_Inland / T_WEGZEIT * 60.
else if T_WEGZEIT eq 0.
   compute speed = -99.
   recode speed (-99 = sysmis).
end if.

variable labels speed 'Daily travel speed in km / hr'.
execute.

do if f42100e = 1.
   compute car_speed = speed.
end if.

variable labels car_speed 'Travel speed only for those with a car always available'.
execute.

means car_speed by age_cohort by gesl
   /cells mean stddev count min max.

T-TEST GROUPS=gesl(1 2)
   /MISSING=ANALYSIS
   /VARIABLES=car_speed
   /CRITERIA=CI(.95).

*calculation of travel speed if in a car with travel time (without waiting).
GET
   FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2010\3_DB_SPSS\wegeinland.sav'.
DATASET NAME wegein2010 WINDOW=FRONT.

do if wmittel = 9 and dauer1 ne 0.
   compute car_only_speed = w_rdist / dauer1 * 60.
end if.

variable labels car_only_speed 'Travel speed for trips in a car'.
execute.

*average speed of all of the car trips for the target person.
DATASET DECLARE speed2010.

AGGREGATE
   /OUTFILE='speed2010'
   /BREAK=HHNR ZIELPNR
   /car_only_speed_mean=MEAN(car_only_speed).
DATASET ACTIVATE speed2010.
SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010car' +
DATASET ACTIVATE cohort2010.
STAR JOIN
/SELECT t0.WP, t0.f41600a, t0.f41600b, t0.f41600c, t0.f41600d, t0.f41600e, t0.f41600f,
t0.f41600g, t0.f42100e, t0.alter, t0.gesl, t0.f20400a, t0.f20400b, t0.rdist_sum_Inland,
t0.e_dauer_sum_Inland, t0.T_WEGE, t0.T_WEGZEIT, t0.hhtyp, t0.F20601, t0.hhgr, t0.income_tercile,
t0.age_cohort, t0.cohort_average_age, t0.pt_sub, t0.pt_sub_1, t0.pt_all, t0.car_speed, t0.car_only_speed_mean
/JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010car speed.sav'
AS t1
ON t0.HHNr=t1.HHNr
AND t0.ZIELPNR=t1.ZIELPNR
/OUTFILE FILE=*.

means car_only_speed_mean by age_cohort by gesl
/cells mean stddev count min max.
T-TEST GROUPS=gesl(1 2)
/MISSING=ANALYSIS
/VARIABLES=car_only_speed_mean
/CRITERIA=CI(.95).

*------------------------------------------------------------------------------------.
*Number of bike/walk trips.
*-----------------------------------------------------------------------------------------------.
GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2010\3_DB_SPSS\wege.sav'.
dataset name walk_bike2010 window=front.
recode wmittel (14 thru 15 = 1) (else = 0) into walk_bike.
execute.

DATASET DECLARE sum_walk_bike.
AGGREGATE
/OUTFILE='sum_walk_bike'
/BREAK=HHNR ZIELPNR
/walk_bike_trips=SUM(walk_bike).
dataset activate sum_walk_bike.
variable labels walk_bike_trips 'Number of trips taken by foot or by bike'.
dataset activate cohort2010.
STAR JOIN
/SELECT t0.WP, t0.f41600a, t0.f41600b, t0.f41600c, t0.f41600d, t0.f41600e, t0.f41600f,
t0.f41600g, t0.f42100e, t0.alter, t0.gesl, t0.f20400a, t0.f20400b, t0.rdist_sum_Inland,
t0.e_dauer_sum_Inland, t0.T_WEGE, t0.T_WEGZEIT, t0.hhtyp, t0.F20601, t0.hhgr, t0.income_tercile,
t0.age_cohort, t0.cohort_average_age, t0.pt_sub, t0.pt_sub_1, t0.pt_all, t0.car_speed, t0.car_only_speed_mean
/t1.walk_bike_trips
/JOIN 'sum_walk_bike' AS t1
ON t0.HHNr=t1.HHNr
AND t0.ZIELPNR=t1.ZIELPNR
/OUTFILE FILE=*.

*-------------------------------------------------- ----------------------------------.
*Collect Weighted Average Data.  
*-------------------------------------------------- ----------------------------------.
weight by WP.
execute.

*Verify: Average car availability by age and gender.
means f42100e by age_cohort by gesl
     /cells = mean stddev count min max.

*PT OWNERSHIP AND CAR AVAILABILITY SPLIT.
*Average daily travel distance (domestic) by age and ownership.
means rdist_sum_Inland by age_cohort by pt_pw
     /cells = mean stddev count min max.

*Average daily trips (domestic) by age and ownership.
means T_WEGE by age_cohort by pt_pw
     /cells = mean stddev count min max.

*Average daily trip time (domestic) by age and ownership.
means T_WEGZEIT by age_cohort by pt_pw
     /cells = mean stddev count min max.

*Average trip speed (domestic) by age and ownership.
means speed by age_cohort by pt_pw
     /cells = mean stddev count min max.

*Average daily trips (domestic) by age and gender.
means T_WEGE by age_cohort by gesl
     /cells = mean stddev count min max.

*Average trip speed (domestic) by age and gender.
means speed by age_cohort by gesl
     /cells = mean stddev count min max.

*Average public transportation subscription by age and gender.
means pt_sub by age_cohort by gesl
     /cells = mean stddev count.

*INCOME TERCILE DATA.
*Average drivers license ownership by age and income.
means f20400a by age_cohort by income_tercile
     /cells = mean stddev count.

*Average car availability by age and income.
means f42100e by age_cohort by income_tercile
     /cells = mean stddev count.

*Average public transportation subscription by age and income.
means pt_sub by age_cohort by income_tercile
     /cells = mean stddev count.

*Average daily travel distance (domestic) by age and income.
means rdist_sum_Inland by age_cohort by income_tercile
/cells = mean stddev count min max.

*Average daily trips (domestic) by age and income.
means T_WEGE by age_cohort by income_tercile
/cells = mean stddev count min max.

*Average daily trip time (domestic) by age and income.
means T_WEGZEIT by age_cohort by income_tercile
/cells = mean stddev count min max.
execute.

*Average trip speed (domestic) by age and income.
means speed by age_cohort by income_tercile
/cells = mean stddev count min max.

*Walk and bike trips by mobility tool ownership.
means walk_bike_trips by pt_pw
/cells = mean stddev count min max.

*Mobility tool ownership split by gender.
do if gesl = 1.
  recode pt_pw (1=1) (2=2) (3=3) (4=4) into pt_pw_gesl.
else if gesl = 2.
  recode pt_pw (1=5) (2=6) (3=7) (4=8) into pt_pw_gesl.
end if.
variable labels pt_pw_gesl 'Mobility tool ownership split by gender'.
value labels pt_pw_gesl
1 'PT and car - male'
2 'car and no PT - male'
3 'PT and no car - male'
4 'no PT no car - male'
5 'PT and car - female'
6 'car and no PT - female'
7 'PT and no car - female'
8 'no PT no car - female'.
execute.
frequencies pt_pw_gesl.

*Test methodology for personal car impact on pt subscription ownership.
*or f41600e = 1.
compute pt_sub_for_analysis_by_car = 0.
if f41600a = 1 pt_sub_for_analysis_by_car = 1.
if f41600d = 1 pt_sub_for_analysis_by_car = 3.
if f41600a = 1 and pt_sub_for_analysis_by_car = 3 pt_sub_for_analysis_by_car = 2.
if f41600b = 1 or f41600e = 1 pt_sub_for_analysis_by_car = 4.
variable labels pt_sub_for_analysis_by_car 'Public transportation subscriptions split into five categories to be used in an analysis comparing car and pt ownership'.
value labels pt_sub_for_analysis_by_car
0 'Kein Abonnement'
1 'Nur Halbtax'
2 'Verbund un Halbtax'
3 'Nur Verbund'
4 ‘GA’.

formats pt_sub_for_analysis_by_car (f8.0).
execute.

*When car is always available.
do if f42100e = 1.
   recode pt_sub_for_analysis_by_car (0 = 0) (1 = 1) (2 = 2) (3 = 3) (4 = 4) into pt_sub_car_avail.
*When car is not always available.
else if f42100e = 0.
   recode pt_sub_for_analysis_by_car (0 = 5) (1 = 6) (2 = 7) (3 = 8) (4 = 9) into pt_sub_car_NOT_avail.
end if.

variable labels pt_sub_car_avail 'Public transportation subscriptions when car is available' / pt_sub_car_NOT_avail 'Public transportation subscriptions when car is NOT available'.
value labels pt_sub_car_avail
  0 'Car Available: Kein Abonnement'
  1 'Car Available: Nur Halbtax'
  2 'Car Available: Verbund un Halbtax'
  3 'Car Available: Nur Verbund'
  4 'Car Available: GA'/
  pt_sub_car_NOT_avail
  5 'Car NOT Available: Kein Abonnement'
  6 'Car NOT Available: Nur Halbtax'
  7 'Car NOT Available: Verbund un Halbtax'
  8 'Car NOT Available: Nur Verbund'
  9 'Car NOT Available: GA'.

formats pt_sub_car_avail (f8.0).
formats pt_sub_car_NOT_avail (f8.0).
execute.

weight by WP.
execute.

frequencies pt_sub_car_avail.
frequencies pt_sub_car_NOT_avail.

A.4 Mobility Analysis 2015

* Encoding: windows-1252.
*-------------------------------------------------- ----------------------------------.
*Age Cohort Brackets.
*-------------------------------------------------- ----------------------------------.
*Uses the alter variable in the zielpersonen file.

*New working file.
*GET
   FILE='P:\_Temp\Ruzsics\MZ2015\code\HHNR.sav'.
*DATASET NAME cohort2015 WINDOW=FRONT.
*SAVE OUTFILE='P:\_Temp\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015cohort.sav'/COMPRESSED.

*Final file.
GET
   FILE='P:\_Temp\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015cohort.sav'.
DATASET NAME cohort2015 WINDOW=FRONT.
weight by WP.
*Add the key variables from zielpersonen.

STAR JOIN

/SELECT t1.WP, t1.alter, t1.fuehrerausweis_pw, t1.fuehrerausweis_motorrad,
  t1.lehnmfahrausweise_pw, t1.lehnmfahrausweise_motorrad, t1.car_available, t1.motor_available,
  t1.pt_sub, t1.gesl, t1.rdist_sum_Inland, t1.etappen_dauer_sum_inland, t1.T_WEGE,
  t1.T_WEGZEIT, t1.oevabo_ga, t1.oevabo_verbund, t1.oevabo_strecke, t1.oevabo_junior_enkel,
  t1.oevabo_anderes, t1.oevabo_h_tax, t1.oevabo_gleis7
/FROM * AS t0
/JOIN 'P:\_TEMP\Ruzsics\MZ2015\res\3_DB_SPSS\CH_SPSS\eng\zielpersonen.sav' AS t1
/ON t0.HHNR=t1.HHNR
/OUTFILE FILE=*.

*Add the key variables from haushalte.

STAR JOIN

/SELECT t0.WP, t0.alter, t0.fuehrerausweis_pw,
  t0.fuehrerausweis_motorrad, t0.lehnmfahrausweise_pw, t0.lehnmfahrausweise_motorrad, t0.car_available,
  t0.motor_available, t0.pt_sub, t0.gesl, t0.rdist_sum_Inland, t0.etappen_dauer_sum_inland,
  t0.T_WEGE, t0.T_WEGZEIT, t0.oevabo_ga, t0.oevabo_verbund, t0.oevabo_strecke,
  t0.oevabo_junior_enkel, t0.oevabo_anderes, t0.oevabo_h_tax, t0.oevabo_gleis7,
  t1.eff_car_own, t1.hhtyp, t1.hhgr, t1.hausalts_einkommen, t1.hausalts_einkommen_I, t1.income_tercile AS income_tercile_imp, t1.WM
/FROM * AS t0
/JOIN 'P:\_TEMP\Ruzsics\MZ2015\res\3_DB_SPSS\CH_SPSS\eng\haushalte.sav' AS t1
/ON t0.HHNR=t1.HHNR
/OUTFILE FILE=*.

*coding to put people in brackets based on their age (in the 10 year brackets used for the Mobility data report).
recode alter (85 thru 104 = 1) (75 thru 84 = 2) (65 thru 74 = 3) (55 thru 64 = 4) (45 thru 54 = 5) (35 thru 44 = 6) (25 thru 34 = 7) (18 thru 24 = 8) (0 thru 17 = 9) into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
 1 '1910 - 1929'
 2 '1930 - 1939'
 3 '1940 - 1949'
 4 '1950 - 1959'
 5 '1960 - 1969'
 6 '1970 - 1979'
 7 '1980 - 1989'
 8 '1990 - 1999'
 9 'under 18 years old and not used for analysis'.
formats age_cohort(f8.0).

*THIS AGE COHORT AVERAGE COMES FROM THE WEIGHTED AVERAGE.
weight by WP.
*finding the average age for each cohort.
means alter by age_cohort
/cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (1 = 87.63) (2=79.35) (3=69.28) (4=59.29) (5=49.51) (6=39.62) (7=29.56) (8=21.08) (9=11.82) into cohort_average_age.
variable labels cohort_average_age 'The average age of all of the individuals in the cohort'.
value labels cohort_average_age
 87.63 '1910 - 1929'
 79.35 '1930 - 1939'
 69.28 '1940 - 1949'
 59.29 '1950 - 1959'
49.51 '1960 - 1969'
39.62 '1970 - 1979'
29.56 '1980 - 1989'
21.08 '1990 - 1999'.
formats cohort_average_age(f8.2).
execute.

*delete the data points where the age is below 18.
SELECT IF age_cohort ne 9.
execute.

*------------------------------------------------------------------------------------.
*Calculate the pt_sub_1 and pt_all variables.
*------------------------------------------------------------------------------------.
*pt_sub_1 includes all public transportation subscriptions (GA, strecke, inter and other)
*pt_all includes anything related to a public transport any variable with oevabo (GA...Halbtax).
*pt_all is the value that was used in Professor Axhausen's analysis of the mobility patterns in Switzerland even though he agrees it shouldn't
be this way.

*pt_sub_1.
if oevabo_ga = 1 pt_sub_1 = 1.
if oevabo_verbund = 1 pt_sub_1 = 1.
if oevabo_strecke = 1 pt_sub_1 = 1.
if oevabo_anderes = 1 pt_sub_1 = 1.
recode pt_sub_1 (sysmis = 0).
variable labels pt_sub_1 'Public transportation subscription including other subscriptions in addition to the pt_sub values'.
formats pt_sub_1 (f8.0).
execute.

*pt_all.
if oevabo_ga = 1 pt_all = 1.
if oevabo_verbund = 1 pt_all = 1.
if oevabo_strecke = 1 pt_all = 1.
if oevabo_junior_enkel = 1 pt_all = 1.
if oevabo_anderes = 1 pt_all = 1.
if oevabo_h_tax = 1 pt_all = 1.
if oevabo_gleis7 = 1 pt_all = 1.
recode pt_all (sysmis = 0).
variable labels pt_all 'Possession of any public transport subscription or discount card'.
formats pt_all (f8.0).
execute.

*Correct Car Availability.
*------------------------------------------------------------------------------------.
do if fuehrerausweis_pw eq 0.
    recode car_available (sysmis = 0).
end if.
frequencies car_available.
execute.

*Possession of car and drivers license.
*------------------------------------------------------------------------------------.
do if sysmis(car_available).
    compute car_and_license= -9.
recode car_and_license (-9 = sysmis).
else if car_available = 0.
    compute car_and_license = 0.
else if fuehrerausweis_pw = 0.
    compute car_and_license = 0.
ELSE.
    compute car_and_license = car_available / fuehrerausweis_pw.
end if.

variable labels car_and_license 'Possession of a drivers license and a car is always available'.
formats car_and_license (f8.0).
execute.

*Classifications for with an without PT sub and passenger car available.
*------------------------------------------------------------------------------------.
if pt_sub eq 1 and car_available eq 1 pt_pw eq 1.
if pt_sub eq 0 and car_available eq 1 pt_pw eq 2.
if pt_sub eq 1 and car_available eq 0 pt_pw eq 3.
if pt_sub eq 0 and car_available eq 0 pt_pw eq 4.

variable labels pt_pw 'Possession of public transport subscriptions and/or passenger vehicles'.
value labels pt_pw
    1 'PT subscription and car available'
    2 'no PT subscription but car available'
    3 'PT subscription but no car available'
    4 'neither of either'.
formats pt_pw (f8.0).
execute.

*Compute Income Terciles without Imputed Income Values.
*------------------------------------------------------------------------------------.
*The original values in the income_tercile variable include imputed income brackets based on social demographics.
*As the method for imputing the income is not very trustworthy, these values should be disregarded.

*Creating a variable for the lower and upper bounds of the income bracket that is provided in the original data.
recode hausalts_einkommen (1 = 1000) (2 = 3000) (3 = 5000) (4 = 7000) (5 = 9000) (6 = 11000) (7 = 13000) (8 = 15000) (9 = 17000) into hausalts_einkommen_mid.
compute eff_income = hausalts_einkommen_mid / sqrt(hhgr).
execute.

*display frequencies to verify.
FREQUENCIES eff_income.

*Setting the income terciles accounting for the number of people per household.
do if missing(eff_income).
    else.
        compute income_tercile = 2.
    recode eff_income (6352 THRU HIGHEST=3) into income_tercile.
    recode eff_income (LOWEST THRU 3600=1) into income_tercile.
end if.
execute.

*English variable labels.
variable labels hausalts_einkommen 'The income terciles'.

76
add value labels income_tercile 1 'First tercile' 2 'Second tercile' 3 'Third tercile'.
formats income_tercile (f2.0).
VARIABLE WIDTH income_tercile (8).
execute.

*display frequencies to verify.
frequencies income_tercile.
execute.

*Income Tercile by Gender.
*to find the percentage of each tercile that is made up of each gender, and to see if there is a difference between genders with respect to income.
do if gesl = 1.
compute income_tercile_gender = income_tercile.
else if gesl = 0.
recode income_tercile (1 = 4) (2 = 5) (3 = 6) into income_tercile_gender.
end if.
execute.
variable labels income_tercile_gender 'Income terciles separated by gender'.
value labels income_tercile_gender
1 'Male: first tercile'
2 'Male: second tercile'
3 'Male: third tercile'
4 'Female: first tercile'
5 'Female: second tercile'
6 'Female: third tercile'.
formats income_tercile_gender (f8.0).
frequencies income_tercile_gender.
execute.

*Motorized vehicles vs Public transport trips per day.
*add the data (edited in Excel)
*I ran a count on the wege file with variable w_verkehrsmittel_agg for how many times per HHNR pt or miv was used.
*this is saved in an SAV file.
STAR JOIN
/SELECT t0.WP, t0.alter, t0.fuehrerausweis_pw, t0.fuehrerausweis_motorrad, 
t0.lehrfahrausweis_pw, t0.lehrfahrausweis_motorrad, t0.car_available, t0.motor_available, 
t0.pt_sub, t0.gesl, t0.ndist_sum_Inland, t0.etappen_dauer_sum_inland, t0.T_WEGE, t0.T_WEGZEIT, 
t0.oevabo_ga, t0.oevabo_verbund, t0.oevabo_strecke, t0.oevabo_junior_enkel, t0.oevabo_anderes, 
t0.oevabo_h_tax, t0.oevabo_gleis7, t0.eff_car_own, t0.hhtyp, t0.hhgr, t0.hausitals_einkommen, 
t0.hausitals_einkommen_1, t0.income_tercile_imp, t0.WM, t0.age_cohort, t0.cohort_average_age, 
t0.pt_sub_1, t0.pt_all, t0.car_and_license, t0.pt_pw, t0.hausitals_einkommen_mid, t0.eff_income, 
t0.income_tercile, t0.income_tercile_gender, t1.pt_trip, t1.miv_trip 
/FROM * AS t0 
JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015 pt and miv + trips.sav' AS t1 
ON t0.HHNR=t1.HHNR 
OUTFILE FILE=*

*Average travel speed.
*
do if etappen_dauer_sum_Inland gt 0.
   compute speed = rdist_sum_Inland / etappen_dauer_sum_Inland *60.
else if etappen_dauer_sum_Inland eq 0.
   compute speed = -99.
   recode speed (-99 = sysmis).
end if.
variable labels speed 'Daily travel speed in km / hr'.
execute.

*Travel speed only with car available.
do if car_available = 1.
   compute car_avail_speed = speed.
end if.
variable labels car_avail_speed 'Travel speed only for those with a car always available'.
execute.
means car_avail_speed by age_cohort by gesl
   /cells mean stddev count min max.

*calculation of travel speed if in a car with travel time (without waiting) and rdist distance measurement.
GET
   FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015wegeinland.sav'.
   DATASET NAME wegein2015 WINDOW=FRONT.
   do if w_verkehrsmittel = 9 and dauer1 ne 0.
      compute car_only_speed = w_rdist / dauer1 *60.
   end if.
variable labels car_only_speed 'Travel speed for trips in a car'.
execute.

DATASET DECLARE carspeed2015.
*average speed of all of the car trips for the target person.
AGGREGATE
   /OUTFILE='carspeed2015'
   /BREAK=HHNR
   /car_only_speed_mean=MEAN(car_only_speed).
SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015car_speed.sav'
   /COMPRESSED.

DATASET ACTIVATE cohort2015.
STAR JOIN
   /SELECT t0.WP, t0.alter, t0.fuehrerausweis_pw, t0.fuehrerausweis_motorrad,
   t0.lehrfahrerausweis_pw, t0.lehrfahrerausweis_motorrad, t0.car_available, t0.motor_available,
   t0.pt_sub, t0.gesl, t0.rdist_sum_Inland, t0.etappen_dauer_sum_inland, t0.T_WEGE, t0.T_WEGZEIT,
   t0.oevabo_ga, t0.oevabo_verbund, t0.oevabo_strecke, t0.oevabo_junior_enkel, t0.oevabo_anderes,
   t0.oevabo_h_tax, t0.oevabo_gleis7, t0.eff_car_ow, t0.hhtyp, t0.hhgr, t0.hausalts_einkommen,
   t0.hausalts_einkommen_1, t0.hausalts_einkommen_ga, t0.income_tercile_imp, t0.WM, t0.age_cohort, t0.cohort_average_age,
   t0.pt_sub_1, t0.pt_all, t0.car_and_license, t0.pt_pw, t0.hausalts_einkommen_mid, t0.eff_income,
   t0.income_tercile, t0.income_tercile_gender, t0.pt_trip, t0.miv_trip, t0.speed, t0.pt_pw_income,
   t0.car_avail_speed, t1.car_only_speed_mean
   /FROM * AS t0
   /JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015car_speed.sav'
   AS t1
   ON t0.HHNR=t1.HHNR
   /OUTFILE FILE=*/.
means car_only_speed_mean by age_cohort by gesl
   /cells mean stddev count min max.
T-TEST GROUPS=gesl(1 2)
/MISSING=ANALYSIS
/VARIABLES=car_only_speed_mean
/CRITERIA=CI(95).

*------------------------------------------------------------------------------------.
*Comparison of car and pt subscription ownership.
*------------------------------------------------------------------------------------.
compute pt_sub_for_analysis_by_car = 0.
if oevabo_h_tax = 1 pt_sub_for_analysis_by_car = 1.
if oevabo_verbund = 1 pt_sub_for_analysis_by_car = 3.
if oevabo_h_tax = 1 and oevabo_verbund = 1 pt_sub_for_analysis_by_car = 2.
if oevabo_ga = 1 pt_sub_for_analysis_by_car = 4.
variable labels pt_sub_for_analysis_by_car 'Public transportation subscriptions split into five categories to be used in an analysis comparing car and pt ownership'.
value labels pt_sub_for_analysis_by_car
0 'Kein Abonnement'
1 'Nur Halbtax'
2 'Verbund un Halbtax'
3 'Nur Verbund'
4 'GA'.
formats pt_sub_for_analysis_by_car (f8.0).
execute.

*When car is always available.
do if car_available = 1.
    recode pt_sub_for_analysis_by_car (0 = 0) (1 = 1) (2 = 2) (3 = 3) (4 = 4) into pt_sub_car_avail.
*When car is not always available.
else if car_available = 0.
    recode pt_sub_for_analysis_by_car (0 = 5) (1 = 6) (2 = 7) (3 = 8) (4 = 9) into pt_sub_car_NOT_avail.
end if.
variable labels pt_sub_car_avail 'Public transportation subscriptions when car is available' / pt_sub_car_NOT_avail 'Public transportation subscriptions when car is NOT available'.
value labels pt_sub_car_avail
0 'Car Available: Kein Abonnement'
1 'Car Available: Nur Halbtax'
2 'Car Available: Verbund un Halbtax'
3 'Car Available: Nur Verbund'
4 'Car Available: GA' / pt_sub_car_NOT_avail
5 'Car NOT Available: Kein Abonnement'
6 'Car NOT Available: Nur Halbtax'
7 'Car NOT Available: Verbund un Halbtax'
8 'Car NOT Available: Nur Verbund'
9 'Car NOT Available: GA'.
formats pt_sub_car_avail (f8.0).
formats pt_sub_car_NOT_avail (f8.0).
execute.

weight by WP.
frequencies pt_sub_car_avail.
frequencies pt_sub_car_NOT_avail.

*Number of bike/walk trips.
*-------------------------------------------------- ----------------------------------.
GET
FILE='P:\_TEMPruzsics\MZ2015\res\3_DB_SPSS\CH_SPSS\eng\wege.sav'.
dataset name walk_bike window=front.

recode w.verkehrsmittel (14 thru 15 = 1) (else = 0) into walk_bike.
execute.

DATASET DECLARE sum_walk_bike.
AGGREGATE
/OUTFILE='sum_walk_bike'
/BREAK=HHNR
/walk_bike_trips=SUM(walk_bike).
dataset activate sum_walk_bike.
variable labels walk_bike_trips 'Number of trips taken by foot or by bike'.
dataset activate cohort2015.
STAR JOIN
/SELECT t0.WP, t0.alter, t0.fuehrerausweis_pw, t0.fuehrerausweis_motorrad,
t0.lehrnfahrausweis_pw, t0.lehrnfahrausweis_motorrad, t0.car_available, t0.motor_available,
t0.pt_sub, t0.gesl, t0.dist_sum_Inland, t0.etappen_dauer_sum_inland, t0.T_WEGE, t0.T_WEGZEIT,
t0.oevabo_ga, t0.oevabo_verbund, t0.oevabo_strecke, t0.oevabo_junior_enkel, t0.oevabo_anderes,
t0.oevabo_h_tax, t0.oevabo_gleis7, t0.eff_car_own, t0.hhtyp, t0.hhgr, t0.hausalts_einkommen,
t0.hausalts_einkommen_I, t0.income_tercile_imp, t0.WM, t0.age_cohort, t0.cohort_average_age,
t0.pt_sub_1, t0.pt_all, t0.car_and_license, t0.pt_pw, t0.hausalts_einkommen_mid, t0.eff_income,
t0.income_tercile, t0.income_tercile_gender, t0.pt_trip, t0.miv_trip, t0.speed, t0.car_avail_speed,
t0.pt_sub_for_analysis_by_car, t0.pt_sub_car_avail, t0.pt_sub_car_NOT_avail, t1.walk_bike_trips
/FROM * AS t0
/JOIN 'sum_walk_bike' AS t1
ON t0.HHNR=t1.HHNR
/OUTFILE FILE='*. 

*-------------------------------------------------------------------------------------------------------------------.
*Collect Weighted Average Data.
*-------------------------------------------------------------------------------------------------------------------.
*weight off.
*execute.
*means WP by age_cohort by gesl
/cells = mean stddev sum count.
weight by WP.
execute.

*Average drivers license ownership.
means fuehrerausweis_pw by age_cohort by gesl
/cells = mean stddev count sum.

*Average car availability by age and gender.
means car_available by age_cohort by gesl
/cells = mean stddev count sum.

*Average public transportation subscription (only) by age and gender.
means pt_sub by age_cohort by gesl
/cells = mean stddev count.
execute.
*Average public transportation subscription or card possession by age and gender.
means pt_all by age_cohort by gesl
   /cells = mean stddev count sum.
execute.

*Average daily travel distance (domestic) by age and gender.
means rdist_sum_Inland by age_cohort by gesl
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and gender.
means T_WEGE by age_cohort by gesl
   /cells = mean stddev count min max.

*Average daily trip time (domestic) by age and gender.
means T_WEGZEIT by age_cohort by gesl
   /cells = mean stddev count min max.

*Average cars per drivers license by age and gender.
means car_and_license by age_cohort by gesl
   /cells = mean stddev count min max.

*Average travel speed by age and gender.
means speed by age_cohort by gesl
   /cells = mean stddev count min max.
execute.

*PT OWNERSHIP AND CAR AVAILABILITY SPLIT.
*Average daily travel distance (domestic) by age and ownership.
means rdist_sum_Inland by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trips (domestic) by age and ownership.
means T_WEGE by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average daily trip time (domestic) by age and ownership.
means T_WEGZEIT by age_cohort by pt_pw
   /cells = mean stddev count min max.

*Average travel speed by age and ownership.
means speed by age_cohort by pt_pw
   /cells = mean stddev count min max.
execute.

*INCOME TERCILE DATA.
*Average drivers license ownership by age and income.
means fuehrerausweis_pw by age_cohort by income_tercile
   /cells = mean stddev count.

*Average car availability by age and income.
means car_available by age_cohort by income_tercile
   /cells = mean stddev count.

*Average public transportation subscription by age and income.
means pt_sub by age_cohort by income_tercile  
/cells = mean stddev count.

*Average daily travel distance (domestic) by age and income. 
means rdist_sum_Inland by age_cohort by income_tercile  
/cells = mean stddev count min max.

*Average daily trips (domestic) by age and income.
means T_WEGER by age_cohort by income_tercile  
/cells = mean stddev count min max.

*Average daily trip time (domestic) by age and income. 
means T_WEGZEIT by age_cohort by income_tercile  
/cells = mean stddev count min max.

*Average travel speed by age and income tercile.
means speed by age_cohort by income_tercile  
/cells = mean stddev count min max.
execute.

*Average travel speed by age and ownership and gender.
means speed by age_cohort by pt_pw by gesl  
/cells = mean stddev count min max.

*Car availability by age and gender.
means car_available by age_cohort by gesl  
/cells = mean stddev count min max.

*PT subscription by age and gender.
means pt_sub by age_cohort by gesl  
/cells = mean stddev count min max.

*trips by PT or car vs ownership. 
*miv trip by tool ownership. 
means miv_trip by pt_pw  
/cells = mean stddev count sum.

*PT trip by tool ownership.
means pt_trip by pt_pw  
/cells = mean stddev count sum.
means walk_bike_trips by pt_pw  
/cells = mean stddev count sum.

*------------------
*Variable Frequencies.
*------------------------------

frequencies fuehrerausweis_pw.
frequencies car_available.
frequencies pt_all.
frequencies T_WEGER.
frequencies rdist_sum_Inland.
frequencies T_WEGZEIT.
means T_WEGZEIT.
frequencies income_tercile.
*Mobility tool ownership split by gender.
do if gesl = 1.
   recode pt_pw (1=1) (2=2) (3=3) (4=4) into pt_pw_gesl.
else if gesl = 2.
   recode pt_pw (1=5) (2=6) (3=7) (4=8) into pt_pw_gesl.
end if.
variable labels pt_pw_gesl 'Mobility tool ownership split by gender'.
value labels pt_pw_gesl
1 'PT and car - male'
2 'car and no PT - male'
3 'PT and no car - male'
4 'no PT no car - male'
5 'PT and car - female'
6 'car and no PT - female'
7 'PT and no car - female'
8 'no PT no car - female'.
execute.
frequencies pt_pw_gesl.

*Mobility tool ownership split by income tercile.
do if income_tercile = 1.
   recode pt_pw (1=1) (2=2) (3=3) (4=4) into pt_pw_income.
else if income_tercile = 2.
   recode pt_pw (1=5) (2=6) (3=7) (4=8) into pt_pw_income.
else if income_tercile = 3.
   recode pt_pw (1=9) (2=10) (3=11) (4=12) into pt_pw_income.
end if.
variable labels pt_pw_income 'Mobility tool ownership split by income tercile'.
value labels pt_pw_income
1 'PT and car - 1'
2 'car and no PT - 1'
3 'PT and no car - 1'
4 'no PT no car - 1'
5 'PT and car - 2'
6 'car and no PT - 2'
7 'PT and no car - 2'
8 'no PT no car - 2'
9 'PT and car - 3'
10 'car and no PT - 3'
11 'PT and no car - 3'
12 'no PT no car - 3'.
execute.
frequencies pt_pw_income.

A.5 Activity Chain Analysis 2005

* Encoding: windows-1252.
*To retrieve the final file.
GET
   FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2005activity.sav'.
DATASET NAME activity2005 WINDOW=FRONT.
weight by WP.
execute.
*Edits from original file follow below.

GET FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2005\MZ205__DB_SPSS\Wegeinland.sav'.
DATASET NAME wegein2005 WINDOW=FRONT.
SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2005 activity calc.sav' /COMPRESSED.

DELETE VARIABLES WP WEGNR DMOD W_X W_Y W_QAL W_BFS W_SPRACHE W_REGION W_KANTON W_AGGLO W_AGGLO_GROESSE W_STRUKTUR W_STRUKTUR_AGG W_STRUKTUR_BFS W платен w_dist_subj w_dist_obj1 wmittela ausnr ausmittel dauer1 wmittel.

DATASET NAME activity_wegein.
SORT CASES BY ZIELPNR (A).
SORT CASES BY HHNR (A).
IF F58 > 1440 F58 = 1440.
IF F514 > 1440 F514 = 1440.
compute dauer2=F514-F58.
recode w_dist_obj2 (-99 = sysmis).
execute.

*copy this file to excel and perform the following calculation in the I column.
*=IF(E1=1;IF(A2=A1;IF(B2=B1;IF(F1 <> -99; C2-D1;););));.
*label the column time_activity.
*save the file as a csv, and then import it into SPSS.
*The imported csv file as an SPSS file.
GET FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2005 importiert csv of activity time.sav'.

DATASET NAME activity_time WINDOW = FRONT.

*simplify the activities into predefined six groups.
recode wzweck1 (sysmis = sysmis) (2 = 1) (3 = 5) (4 = 2) (6 thru 7 = 4) (8 = 3) (else = 6) into activity_6.
execute.

*sum the activity times and travel times for each person and each group.
DATASET DECLARE time_split.
AGGREGATE /OUTFILE='time_split'
/BREAK=HHNR ZIELPNR activity_6
/time_activity_sum 'total time spent at the activity'=SUM(time_activity)
/w_dist_sum 'total distance travelled to reach activity'=SUM(w_dist_obj2).

DATASET ACTIVATE activity_time.
DATASET DECLARE travel_time_2005.
AGGREGATE
/OUTFILE='travel_time_2005'
/BREAK=HHNR ZIELPNR
/travel_time_sum 'sum of the travel time including waiting and transfer times'=SUM(dauer2).
dataset activate travel_time_2005.
compute travel_time = travel_time_sum / 60.
execute.
compute work_time = time_activity_sum / 60.
else if activity_6 = 2.
    compute shopping_time = time_activity_sum / 60.
else if activity_6 = 3.
    compute leisure_time = time_activity_sum / 60.
else if activity_6 = 4.
    compute business_time = time_activity_sum / 60.
else if activity_6 = 5.
    compute education_time = time_activity_sum / 60.
else if activity_6 = 6.
    compute other_time = time_activity_sum / 60.
end if.

variable labels work_time 'time spent at work (hours)'
shopping_time 'time spent shopping (hours)'
leisure_time 'time spent in leisure (hours)'
business_time 'time spent in business (hours)'
education_time 'time spent in education (hours)'
other_time 'time spent at other activities (hours)'.
execute.

*split up distance travelled for each activity.
compute work_dist = 0.
compute shopping_dist = 0.
compute leisure_dist = 0.
compute business_dist = 0.
compute education_dist = 0.
compute other_dist = 0.

variable labels work_dist 'distance travelled for work (km)'
shopping_dist 'distance travelled for shopping (km)'
leisure_dist 'distance travelled for leisure (km)'
business_dist 'distance travelled for business (km)'
education_dist 'distance travelled for education (km)'
other_dist 'distance travelled for other (km)'.
execute.
*regroup the data so that each HHNR takes up one line of data and the separate activity times are included for each person.

DATASET DECLARE activity_time_HHNR.

AGGREGATE
   /OUTFILE='activity_time_HHNR'
   /BREAK=HHNR ZIELPNR
   /work_time 'time spent at work'=SUM(work_time)
   /shopping_time 'time spent shopping'=SUM(shopping_time)
   /leisure_time 'time spent in leisure'=SUM(leisure_time)
   /business_time 'time spent in business activities'=SUM(business_time)
   /education_time 'time spent in education'=SUM(education_time)
   /other_time 'time spent in other activities'=SUM(other_time)
   /work_dist 'distance travelled for work activities'=SUM(work_dist)
   /shopping_dist 'distance travelled for shopping activities'=SUM(shopping_dist)
   /leisure_dist 'distance travelled for leisure activities'=SUM(leisure_dist)
   /business_dist 'distance travelled for business activities'=SUM(business_dist)
   /education_dist 'distance travelled for education activities'=SUM(education_dist)
   /other_dist 'distance travelled for other activities'=SUM(other_dist).

dataset activate activity_time_HHNR.

STAR JOIN
   /SELECT t0.work_time, t0.shopping_time, t0.leisure_time, t0.business_time, t0.education_time,
   t0.other_time, t0.work_dist, t0.shopping_dist, t0.leisure_dist, t0.business_dist, t0.education_dist,
   t0.other_dist, t1.F23B, t1.F23C, t1.T_ARBEIT, t1.T_AUSBILDUNG, t1.T_EINKAUF, t1.T_NUTZ,
   t1.T_FREIZEIT, t1.T_BEGLEIT
   /FROM * AS t0
   /JOIN 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2005\MZ05_Datenbank_CH\DB_SPSS\zielpersonen.sav' AS t1
   ON t0.HHNR=t1.HHNR AND t0.ZIELPNR=t1.ZIELPNR
   /OUTFILE FILE=*

STAR JOIN
   /SELECT t0.work_time, t0.shopping_time, t0.leisure_time, t0.business_time, t0.education_time,
   t0.other_time, t0.work_dist, t0.shopping_dist, t0.leisure_dist, t0.business_dist, t0.education_dist,
   t0.other_dist, t1.travel_time
   /FROM * AS t0
   /JOIN 'travel_time_2005' AS t1
   ON t0.HHNR=t1.HHNR AND t0.ZIELPNR=t1.ZIELPNR
   /OUTFILE FILE=*

*coding to put people in ten year cohort brackets based on their age.
recode F23B (75 thru 94 = 1) (65 thru 74 = 2) (55 thru 64 = 3) (45 thru 54 = 4) (35 thru 44 = 5) (25 thru 34 = 6) (15 thru 24 = 7) (5 thru 14 = 8)
into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
  1 '1910 - 1929'
  2 '1930 - 1939'
  3 '1940 - 1949'
  4 '1950 - 1959'
  5 '1960 - 1969'
  6 '1970 - 1979'
  7 '1980 - 1989'
  8 '1990 - 1999'
  9 '2000 - 2009'.
formats age_cohort(f8.0).
*age cohort average age calculation.
weight by WP.
execute.

*finding the average age for each cohort.
means F23B by age_cohort
    /cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (1 = 79.76) (2=69.25) (3=59.25) (4=49.21) (5=39.57) (6=29.75) (7=19.38) (8=10.22) into cohort_average_age.
variable labels cohort_average_age 'The average age of all of the individuals in the cohort'.
value labels cohort_average_age
    79.76 '1910 - 1929'
    69.25 '1930 - 1939'
    59.25 '1940 - 1949'
    49.21 '1950 - 1959'
    39.57 '1960 - 1969'
    29.75 '1970 - 1979'
    19.38 '1980 - 1989'
    10.22 '1990 - 1999'.
execute.

*compute the total time spent at home and out of home.
compute out_time = work_time + shopping_time + leisure_time + business_time + education_time + other_time + travel_time.
compute home_time = 24 - out_time.
compute activity_time = out_time - work_time.
variable labels out_time 'Time in hours spent out of the home in a day' home_time 'Time in hours spent at home in a day' activity_time 'Time in hours spent in total at activities in a day'.
execute.

*average time at work.
means work_time by age_cohort
    /cells = mean stddev count min max.

*average time shopping.
means shopping_time by age_cohort
    /cells = mean stddev count min max.

*average time at leisure.
means leisure_time by age_cohort
    /cells = mean stddev count min max.

*average time at business.
means business_time by age_cohort
    /cells = mean stddev count min max.

*average time at education.
means education_time by age_cohort
    /cells = mean stddev count min max.

*average time at other.
means other_time by age_cohort
    /cells = mean stddev count min max.

*average distance at work.
means work_dist by age_coHORT
  /cells = mean stddev count min max.

*average distance shopping.
means shopping_dist by age_coHORT
  /cells = mean stddev count min max.

*average distance at leisure.
means leisure_dist by age_coHORT
  /cells = mean stddev count min max.

*average distance at business.
means business_dist by age_coHORT
  /cells = mean stddev count min max.

*average distance at education.
means education_dist by age_coHORT
  /cells = mean stddev count min max.

*average distance at other.
means other_dist by age_coHORT
  /cells = mean stddev count min max.

*average number of journeys for work.
means T_ARBEIT by age_coHORT
  /cells = mean stddev count min max.

*average number of journeys for shopping.
means T_EINKAUF by age_coHORT
  /cells = mean stddev count min max.

*average number of journeys for leisure.
means T_FREIZEIT by age_coHORT
  /cells = mean stddev count min max.

*average number of journeys for business.
means T_NUTZ by age_coHORT
  /cells = mean stddev count min max.

*average number of journeys for education.
means T_AUSBILDUNG by age_coHORT
  /cells = mean stddev count min max.

*average number of journeys for other.
means T_BEGLEIT by age_coHORT
  /cells = mean stddev count min max.

*average time at home.
means home_time by age_coHORT
  /cells = mean stddev count min max.

*average time out of home.
means out_time by age_coHORT
  /cells = mean stddev count min max.

*average time out of home at activities.
means activity_time by age_cohort
   /cells = mean stddev count min max.

*same analysis with gender also included.
*average time at work.
means work_time by age_cohort by F23C
   /cells = mean stddev count min max.

*average time shopping.
means shopping_time by age_cohort by F23C
   /cells = mean stddev count min max.

*average time at leisure.
means leisure_time by age_cohort by F23C
   /cells = mean stddev count min max.

*average time at business.
means business_time by age_cohort by F23C
   /cells = mean stddev count min max.

*average time at education.
means education_time by age_cohort by F23C
   /cells = mean stddev count min max.

*average time at other.
means other_time by age_cohort by F23C
   /cells = mean stddev count min max.

*average distance at work.
means work_dist by age_cohort by F23C
   /cells = mean stddev count min max.

*average distance shopping.
means shopping_dist by age_cohort by F23C
   /cells = mean stddev count min max.

*average distance at leisure.
means leisure_dist by age_cohort by F23C
   /cells = mean stddev count min max.

*average distance at business.
means business_dist by age_cohort by F23C
   /cells = mean stddev count min max.

*average distance at education.
means education_dist by age_cohort by F23C
   /cells = mean stddev count min max.
*average distance at other.
means other_dist by age_cohort by F23C
/cells = mean stddev count min max.

*average number of journeys for work.
means T_ARBEIT by age_cohort by F23C
/cells = mean stddev count min max.

*average number of journeys for shopping.
means T_EINKAUF by age_cohort by F23C
/cells = mean stddev count min max.

*average number of journeys for leisure.
means T_FREIZEIT by age_cohort by F23C
/cells = mean stddev count min max.

*average number of journeys for business.
means T_NUTZ by age_cohort by F23C
/cells = mean stddev count min max.

*average number of journeys for education.
means T_AUSBILDUNG by age_cohort by F23C
/cells = mean stddev count min max.

*average number of journeys for other.
means T_BEGLEIT by age_cohort by F23C
/cells = mean stddev count min max.

*average time at home.
means home_time by age_cohort by F23C
/cells = mean stddev count min max.

*average time out of home.
means out_time by age_cohort by F23C
/cells = mean stddev count min max.

*average time out of home at activities.
means activity_time by age_cohort by F23C
/cells = mean stddev count min max.

A.6 Activity Chain Analysis 2010

* Encoding: windows-1252.
*To retrieve the final file.
GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010activity.sav'.
DATASET NAME activity2010 WINDOW=FRONT.

*Edits from original file follow below.
GET
   FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2010\DB_SPSS\wegeinland.sav'.
DATASET NAME wegein2010 WINDOW=FRONT.

SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010 '+
   'activity calc.sav' /COMPRESSED.

DELETE VARIABLES WP WEGNR DMOD pseudo W_X W_Y w_x_CH1903 w_y_CH1903 W_QAL W_BFS
   W_SPRACHE W_REGION W_KANTON W_AGGLO W_AGGLO_GROESSE W_STRUKTUR W_STRUKTUR_AGG
   w_struktur_bfs9 S_X S_Y S_X_CH1903 S_Y_CH1903 S_QAL S_BFS S_PLZ S_Ort S_Str S_hnr S_SPRACHE
   S_REGION S_KANTON S_NUTS3 S_LND S_AGGLO S_AGGLO_GROESSE S_STRUKTUR S_STRUKTUR_AGG
   S_struktur_bfs9 Z_X Z_Y Z_X_CH1903 Z_Y_CH1903 Z_QAL Z_BFS Z_PLZ Z_Ort Z_Str Z_hnr Z_SPRACHE Z_REGION
   Z_KANTON Z_NUTS3 Z_LND Z_AGGLO Z_AGGLO_GROESSE Z_STRUKTUR Z_STRUKTUR_AGG Z_struktur_bfs9
   f80700a f80700b f80700c f51700 f51800a f51800b f51800c f51900 f52000a f52000b f52000c
   f52000d dauer1 w_etappen w_dist_subj w_luftlin wmittela w_dist_obj2 ausnr.

DATASET NAME activity_wegein.

*remove the trips that take place after midnight of the specified day.
IF F51100 > 1440 F51100 = 1440.
IF F51400 > 1440 F51400 = 1440.
compute dauer2=F51400-F51100.

*copy this file to excel and perform the following calculation in the I column.
*=IF(G1=1;IF(A2=A1;IF(B2=B1;C2-D1;););).
*label the column time_activity.
*save the file as a csv, and then import it into SPSS.
*The imported csv file as an SPSS file.
GET
   FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2010 imported csv of activity time.sav'.

DATASET NAME activity_time WINDOW=FRONT.

*simplify the activities into predefined six groups.
recode wzweck1 (sysmis=sysmis) (2=1) (3=5) (4=2) (6 thru 7=4) (8=3) (else=6) into activity_6.
execute.

*sum the activity times and travel times for each person and each group.
DATASET DECLARE time_distance_split.
AGGREGATE
   /OUTFILE='time_distance_split'
   /BREAK=HHNR ZIELPNR activity_6
   /time_activity_sum 'total time spent at the activity'=SUM(time_activity)
   /w_rdist_sum 'total distance travelled to reach activity'=SUM(w_rdist).
DATASET ACTIVATE activity_time.

DATASET DECLARE travel_time_2010.
AGGREGATE
   /OUTFILE='travel_time_2010'
   /BREAK=HHNR ZIELPNR
   /travel_time_sum 'sum of the travel time including waiting and transfer times'=SUM(dauer2).
dataset activate travel_time_2010.
compute travel_time = travel_time_sum / 60.
execute.

dataset activate time_distance_split.
compute work_time = 0.
compute shopping_time = 0.
compute leisure_time = 0.
compute business_time = 0.
compute education_time = 0.
compute other_time = 0.
do if activity_6 = 1.
   compute work_time = time_activity_sum / 60.
else if activity_6 = 2.
   compute shopping_time = time_activity_sum / 60.
else if activity_6 = 3.
   compute leisure_time = time_activity_sum / 60.
else if activity_6 = 4.
   compute business_time = time_activity_sum / 60.
else if activity_6 = 5.
   compute education_time = time_activity_sum / 60.
else if activity_6 = 6.
   compute other_time = time_activity_sum / 60.
end if.
variable labels work_time 'time spent at work (hours)'
shopping_time 'time spent shopping (hours)'
leisure_time 'time spent in leisure (hours)'
business_time 'time spent in business (hours)'
education_time 'time spent in education (hours)'
other_time 'time spent at other activities (hours)'.
execute.

*split up distance travelled for each activity.
compute work_dist = 0.
compute shopping_dist = 0.
compute leisure_dist = 0.
compute business_dist = 0.
compute education_dist = 0.
compute other_dist = 0.
do if activity_6 = 1.
   compute work_dist = w_rdist_sum.
else if activity_6 = 2.
   compute shopping_dist = w_rdist_sum.
else if activity_6 = 3.
   compute leisure_dist = w_rdist_sum.
else if activity_6 = 4.
   compute business_dist = w_rdist_sum.
else if activity_6 = 5.
   compute education_dist = w_rdist_sum.
else if activity_6 = 6.
   compute other_dist = w_rdist_sum.
end if.
variable labels work_dist 'distance travelled for work (km)'
shopping_dist 'distance travelled for shopping (km)'
leisure_dist 'distance travelled for leisure (km)'
business_dist 'distance travelled for business (km)'
education_dist 'distance travelled for education (km)'


other_dist 'distance travelled for other (km)'.
execute.

*regroup the data so that each HHNR takes up one line of data and the separate activity times are included for each person.

DATASET DECLARE activity_time_HHNR.

AGGREGATE
/OUTFILE='activity_time_HHNR'
/BREAK=HHNR ZIELPNR
/work_time 'time spent at work'=SUM(work_time)
/shopping_time 'time spent shopping'=SUM(shopping_time)
/leisure_time 'time spent in leisure'=SUM(leisure_time)
/business_time 'time spent in business activities'=SUM(business_time)
/education_time 'time spent in education'=SUM(education_time)
/other_time 'time spent in other activities'=SUM(other_time)
/work_dist 'distance travelled for work activities'=SUM(work_dist)
/shopping_dist 'distance travelled for shopping activities'=SUM(shopping_dist)
/leisure_dist 'distance travelled for leisure activities'=SUM(leisure_dist)
/business_dist 'distance travelled for business activities'=SUM(business_dist)
/education_dist 'distance travelled for education'=SUM(education_dist)
/other_dist 'distance travelled for other activities'=SUM(other_dist).

dataset activate activity_time_HHNR.

STAR JOIN
/SELECT t0.work_time, t0.shopping_time, t0.leisure_time, t0.business_time, t0.education_time, t0.other_time, t0.work_dist, t0.shopping_dist, t0.leisure_dist, t0.business_dist, t0.education_dist, t0.other_dist, t0.WP, t0.alter, t0.gesl, t0.T_ARBREIT, t0.T_AUSBILDUNG, t0.T_EINKAUF, t0.T_NUTZ, t0.T_FREIZEIT, t0.T_BEGLEIT
/FROM * AS t0
/JION 'P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\Past Microcensuses\MZ2010\3_DB_SPSS\zielpersonen.sav' AS t1
 ON t0.HHNR=t1.HHNR and t0.ZIELPNR=t1.ZIELPNR
/OUTFILE FILE=*.

STAR JOIN
/SELECT t0.work_time, t0.shopping_time, t0.leisure_time, t0.business_time, t0.education_time, t0.other_time, t0.work_dist, t0.shopping_dist, t0.leisure_dist, t0.business_dist, t0.education_dist, t0.other_dist, t0.WP, t0.alter, t0.gesl, t0.T_ARBREIT, t0.T_AUSBILDUNG, t0.T_EINKAUF, t0.T_NUTZ, t0.T_FREIZEIT, t0.T_BEGLEIT, t0.travel_time
/FROM * AS t0
/JION 'travel_time_2010' AS t1
 ON t0.HHNR=t1.HHNR and t0.ZIELPNR=t1.ZIELPNR
/OUTFILE FILE=*.

*coding to put people in ten year cohort brackets based on their age.
recode alter (80 thru 99 = 1) (70 thru 79 = 2) (60 thru 69 = 3) (50 thru 59 = 4) (40 thru 49 = 5) (30 thru 39 = 6) (20 thru 29 = 7) (10 thru 19 = 8) (0 thru 9 = 9) into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
1 '1910 - 1929'
2 '1930 - 1939'
3 '1940 - 1949'
4 '1950 - 1959'
5 '1960 - 1969'
6 '1970 - 1979'
7 '1980 - 1989'
8 '1990 - 1999'
*age cohort average age calculation.
weight by WP.
execute.
*finding the average age for each cohort.
means alter by age_cohort
   /cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (1=83.71) (2=74.13) (3=64.23) (4=54.26) (5=44.41) (6=34.70) (7=24.58) (8=14.75) (9=7.50) into cohort_average_age.
variable labels cohort_average_age 'The average age of all of the individuals in the cohort'.
value labels cohort_average_age
83.71 '1910 - 1929'
74.13 '1930 - 1939'
64.23 '1940 - 1949'
54.26 '1950 - 1959'
44.41 '1960 - 1969'
34.70 '1970 - 1979'
24.58 '1980 - 1989'
14.75 '1990 - 1999'
7.50 '2000 - 2009'.
execute.

*compute the total time spent at home and out of home.
compute out_time = work_time + shopping_time + leisure_time + business_time + education_time + other_time + travel_time.
compute home_time = 24 - out_time.
compute activity_time = out_time - work_time.
variable labels out_time 'Time in hours spent out of the home in a day' home_time 'Time in hours spent at home in a day' activity_time 'Time in hours spent in total at activities in a day'.
execute.

*means the distance is only measured by the individuals who took the trip and not the whole population.
recode work_dist (0=sysmis).
recode shopping_dist (0=sysmis).
recode leisure_dist (0=sysmis).
recode business_dist (0=sysmis).
recode education_dist (0=sysmis).
recode other_dist (0=sysmis).
execute.

*average time at work.
means work_time by age_cohort
   /cells = mean stddev count min max.

*average time shopping.
means shopping_time by age_cohort
   /cells = mean stddev count min max.

*average time at leisure.
means leisure_time by age_cohort
   /cells = mean stddev count min max.
*average time at business.
means business_time by age_cohort
/cells = mean stddev count min max.

*average time at education.
means education_time by age_cohort
/cells = mean stddev count min max.

*average time at other.
means other_time by age_cohort
/cells = mean stddev count min max.

*average distance at work.
means work_dist by age_cohort
/cells = mean stddev count min max.

*average distance at business.
means business_dist by age_cohort
/cells = mean stddev count min max.

*average distance at leisure.
means leisure_dist by age_cohort
/cells = mean stddev count min max.

*average distance at education.
means education_dist by age_cohort
/cells = mean stddev count min max.

*average distance at other.
means other_dist by age_cohort
/cells = mean stddev count min max.

*average number of journeys for work.
means T_ARBEIT by age_cohort
/cells = mean stddev count min max.

*average number of journeys for leisure.
means T_FREIZEIT by age_cohort
/cells = mean stddev count min max.

*average number of journeys for business.
means T_NUTZ by age_cohort
/cells = mean stddev count min max.

*average number of journeys for education.
means T_AUSBILDUNG by age_cohort
/cells = mean stddev count min max.
*average number of journeys for other. 
   means T_BEGLEIT by age_cohort 
   /cells = mean stddev count min max.

*average time at home. 
   means home_time by age_cohort 
   /cells = mean stddev count min max.

*average time out of home. 
   means out_time by age_cohort 
   /cells = mean stddev count min max.

*average time out of home at activities. 
   means activity_time by age_cohort 
   /cells = mean stddev count min max.

*activities also plotted by gender. 
*average time at work. 
   means work_time by age_cohort by gesl 
   /cells = mean stddev count min max.

*average time shopping. 
   means shopping_time by age_cohort by gesl 
   /cells = mean stddev count min max.

*average time at leisure. 
   means leisure_time by age_cohort by gesl 
   /cells = mean stddev count min max.

*average time at business. 
   means business_time by age_cohort by gesl 
   /cells = mean stddev count min max.

*average time at education. 
   means education_time by age_cohort by gesl 
   /cells = mean stddev count min max.

*average time at other. 
   means other_time by age_cohort by gesl 
   /cells = mean stddev count min max.

*average distance at work. 
   means work_dist by age_cohort by gesl 
   /cells = mean stddev count min max.

*average distance shopping. 
   means shopping_dist by age_cohort by gesl 
   /cells = mean stddev count min max.

*average distance at leisure. 
   means leisure_dist by age_cohort by gesl 
   /cells = mean stddev count min max.

*average distance at business. 
   means business_dist by age_cohort by gesl 
   /cells = mean stddev count min max.
*average distance at education.
means education_dist by age_cohort by gesl
/cells = mean stddev count min max.

*average distance at other.
means other_dist by age_cohort by gesl
/cells = mean stddev count min max.

*average number of journeys for work.
means T_ARBEIT by age_cohort by gesl
/cells = mean stddev count min max.

*average number of journeys for shopping.
means T_EINKAUF by age_cohort by gesl
/cells = mean stddev count min max.

*average number of journeys for leisure.
means T_FREIZEIT by age_cohort by gesl
/cells = mean stddev count min max.

*average number of journeys for business.
means T_NUTZ by age_cohort by gesl
/cells = mean stddev count min max.

*average number of journeys for education.
means T_AUSBILDUNG by age_cohort by gesl
/cells = mean stddev count min max.

*average number of journeys for other.
means T_BEGLEIT by age_cohort by gesl
/cells = mean stddev count min max.

*average time at home.
means home_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time out of home.
means out_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time out of home at activities.
means activity_time by age_cohort by gesl
/cells = mean stddev count min max.

**A.7 Activity Chain Analysis 2015**

* Encoding: windows-1252.
* To retrieve the final file.
  GET
    FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015activity.sav'.
    DATASET NAME activity2015 WINDOW=FRONT.
    weight by WP.
  execute.
*Edits from original file follow below.

GET
FILE='P:\_TEMP\Ruzsics\MZ2015\res\3_DB_SPSS\CH_SPSS\eng\wegeinland.sav'.
DATASET NAME wegein2015 WINDOW=FRONT.

SAVE OUTFILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015 '+
'activity calc.sav'
/COMPRESSED.

DATASET NAME activity_wegein.
*remove the trips that take place after midnight of the specified day.
IF abfahrtszeit > 1440 abfahrtszeit = 1440.
IF ankuftszeit > 1440 ankuftszeit = 1440.
compute dauer2 = ankuftszeit - abfahrtszeit.
execute.

*copy this file to excel and perform the following calculation in the L column.
*=IF(H1=1;IF(A2=A1;F2-G1;);).
*label the column time_activity.
*save the file as a csv, and then import it into SPSS.
*The imported csv file as an SPSS file.
GET
FILE='P:\_TEMP\Smith\Update Mobility Patterns\Relevant to Mobility Update\mz2015 imported csv of activity time.sav'.
DATASET NAME activity_time WINDOW = FRONT.

*simplify the activities into predefined six groups.
recode w_zweck_weg_3 (sysmis = sysmis) (2 = 1) (3 = 5) (4 = 2) (6 thru 7 = 4) (8 = 3) (else = 6) into activity_6.
execute.

*sum the activity times and travel times and travel distance for each person and each group.
DATASET DECLARE time_distance_split.
AGGREGATE
/OUTFILE=time_distance_split
/BREAK=HHNR activity_6
/time_activity_sum=SUM(time_activity)
/w_rdist_sum=SUM(w_rdist).

DATASET ACTIVATE activity_time.
DATASET DECLARE travel_time_2015.
AGGREGATE
/OUTFILE='travel_time_2015' 
/BREAK=HHNR 
/travel_time_sum 'sum of the travel time including waiting and transfer times'=SUM(dauer2).
dataset activate travel_time_2015.
compute travel_time = travel_time_sum / 60.
execute.

*split up time spent at each activity.
dataset activate time_distance_split.
compute work_time = 0.
compute shopping_time = 0.
compute leisure_time = 0.
compute business_time = 0.
compute education_time = 0.
compute other_time = 0.
do if activity_6 = 1.
   compute work_time = time_activity_sum / 60.
else if activity_6 = 2.
   compute shopping_time = time_activity_sum / 60.
else if activity_6 = 3.
   compute leisure_time = time_activity_sum / 60.
else if activity_6 = 4.
   compute business_time = time_activity_sum / 60.
else if activity_6 = 5.
   compute education_time = time_activity_sum / 60.
else if activity_6 = 6.
   compute other_time = time_activity_sum / 60.
end if.
variable labels work_time 'time spent at work (hours)'
shopping_time 'time spent shopping (hours)'
leisure_time 'time spent in leisure (hours)'
business_time 'time spent in business (hours)'
education_time 'time spent in education (hours)'
other_time 'time spent at other activities (hours)'.
execute.

*split up distance travelled for each activity.
compute work_dist = 0.
compute shopping_dist = 0.
compute leisure_dist = 0.
compute business_dist = 0.
compute education_dist = 0.
compute other_dist = 0.
do if activity_6 = 1.
   compute work_dist = w_rdist_sum.
else if activity_6 = 2.
   compute shopping_dist = w_rdist_sum.
else if activity_6 = 3.
   compute leisure_dist = w_rdist_sum.
else if activity_6 = 4.
   compute business_dist = w_rdist_sum.
else if activity_6 = 5.
   compute education_dist = w_rdist_sum.
else if activity_6 = 6.
   compute other_dist = w_rdist_sum.

compute other_dist = w_rdist_sum.
end if.
variable labels work_dist 'distance travelled for work (km)'
  shopping_dist 'distance travelled for shopping (km)'
  leisure_dist 'distance travelled for leisure (km)'
  business_dist 'distance travelled for business (km)'
  education_dist 'distance travelled for education (km)'
  other_dist 'distance travelled for other (km)'.
execute.

*regroup the data so that each HHNR takes up one line of data and the separate activity times are included for each person.
DATASET DECLARE activity_time_HHNR.
AGGREGATE
/OUTFILE='activity_time_HHNR'
/BREAK=HHNR
/work_time 'time spent at work'=SUM(work_time)
/shopping_time 'time spent shopping'=SUM(shopping_time)
/leisure_time 'time spent in leisure'=SUM(leisure_time)
/business_time 'time spent in business activities'=SUM(business_time)
/education_time 'time spent in education'=SUM(education_time)
/other_time 'time spent in other activities'=SUM(other_time)
/work_dist 'distance travelled for work activities'=SUM(work_dist)
/shopping_dist 'distance travelled for shopping activities'=SUM(shopping_dist)
/leisure_dist 'distance travelled for leisure activities'=SUM(leisure_dist)
/business_dist 'distance travelled for business activities'=SUM(business_dist)
/education_dist 'distance travelled for education'=SUM(education_dist)
/other_dist 'distance travelled for other activities'=SUM(other_dist).
dataset activate activity_time_HHNR.
STAR JOIN
/SELECT t0.work_time, t0.shopping_time, t0.leisure_time, t0.business_time, t0.education_time,
  t0.other_time, t0.work_dist, t0.shopping_dist, t0.leisure_dist, t0.business_dist, t0.education_dist,
  t0.other_dist, t1.WP, t1.alter, t1.gesl, t1.T_ARBEIT, t1.T_AUSBILDUNG, t1.T_EINKAUF, t1.T_NUTZ,
  t1.T_FREIZEIT, t1.T_BEGLEIT
/FROM * AS t0
/JOIN 'P:\_TEMP\Ruzsics\MZ2015\ees\3_DB_SPSS\CH_SPSS\Eng\zielpersonen.sav' AS t1
  ON t0.HHNR=t1.HHNR
/OUTFILE FILE=*.

*coding to put people in ten year cohort brackets based on their age.
recode alter (85 thru 104 = 1) (75 thru 84 = 2) (65 thru 74 = 3) (55 thru 64 = 4) (45 thru 54 = 5) (35 thru 44 = 6) (25 thru 34 = 7) (15 thru 24 = 8) (6 thru 14 = 9) into age_cohort.
variable labels age_cohort 'Age split into brackets for analysis based on the decade of birth'.
value labels age_cohort
  1 '1910 - 1929'
  2 '1930 - 1939'
formats age_cohort(f8.0).

*age cohort average age calculation.
weight by WP. 
execute.
*finding the average age for each cohort.
means alter by age_cohort
   /cells = mean stddev min max.
execute.

*manually add in the average ages that are provided by the above table.
recode age_cohort (1 = 87.48) (2=79.24) (3=69.23) (4=59.24) (5=49.51) (6=39.62) (7=29.53) (8=19.76) (9=10.25) into cohort_average_age.
value labels cohort_average_age 'The average age of all of the individuals in the cohort'.

compute the total time spent at home and out of home.
compute out_time = work_time + shopping_time + leisure_time + business_time + education_time + other_time + travel_time.
compute home_time = 24 - out_time.
compute activity_time = out_time - work_time.
variable labels out_time 'Time in hours spent out of the home in a day' home_time 'Time in hours spent at home in a day' activity_time 'Time in hours spent in total at activities in a day'.
execute.

*means the distance is only measured by the individuals who took the trip and not the whole population.
recode work_dist (0 = sysmis).
recode shopping_dist (0 = sysmis).
recode leisure_dist (0 = sysmis).
recode business_dist (0 = sysmis).
recode education_dist (0 = sysmis).
recode other_dist (0 = sysmis). 
execute.

*average time at work.
means work_time by age_cohort
   /cells = mean stddev count min max.

*average time shopping.
means shopping_time by age_cohort
*average time at leisure.
means leisure_time by age_cohort
/cells = mean stddev count min max.

*average time at business.
means business_time by age_cohort
/cells = mean stddev count min max.

*average time at education.
means education_time by age_cohort
/cells = mean stddev count min max.

*average time at other.
means other_time by age_cohort
/cells = mean stddev count min max.

*average distance at work.
means work_dist by age_cohort
/cells = mean stddev count min max.

*average distance shopping.
means shopping_dist by age_cohort
/cells = mean stddev count min max.

*average distance at leisure.
means leisure_dist by age_cohort
/cells = mean stddev count min max.

*average distance at business.
means business_dist by age_cohort
/cells = mean stddev count min max.

*average distance at education.
means education_dist by age_cohort
/cells = mean stddev count min max.

*average distance at other.
means other_dist by age_cohort
/cells = mean stddev count min max.

*average number of journeys for work.
means T_ARBEIT by age_cohort
/cells = mean stddev count min max.

*average number of journeys for shopping.
means T_EINKAUF by age_cohort
/cells = mean stddev count min max.

*average number of journeys for leisure.
means T_FREIZEIT by age_cohort
/cells = mean stddev count min max.

*average number of journeys for business.
means T_NUTZ by age_cohort
/cells = mean stddev count min max.

*average number of journeys for education.
means T_AUSBILDUNG by age_cohort
/cells = mean stddev count min max.

*average number of journeys for other.
means T_BEGLEIT by age_cohort
/cells = mean stddev count min max.

*average time at home.
means home_time by age_cohort
/cells = mean stddev count min max.

*average time out of home.
means out_time by age_cohort
/cells = mean stddev count min max.

*average time out of home at activities.
means activity_time by age_cohort
/cells = mean stddev count min max.

*data plotted by gender as well.
*average time at work.
means work_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time shopping.
means shopping_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time at leisure.
means leisure_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time at business.
means business_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time at education.
means education_time by age_cohort by gesl
/cells = mean stddev count min max.

*average time at other.
means other_time by age_cohort by gesl
/cells = mean stddev count min max.

*average distance at work.
means work_dist by age_cohort by gesl
*average distance shopping.
means shopping_dist by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average distance at leisure.
means leisure_dist by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average distance at business.
means business_dist by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average distance at education.
means education_dist by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average distance at other.
means other_dist by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average number of journeys for work.
means T_ARBEIT by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average number of journeys for shopping.
means T_EINKAUF by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average number of journeys for leisure.
means T_FREIZEIT by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average number of journeys for business.
means T_NUTZ by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average number of journeys for education.
means T_AUSBILDUNG by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average number of journeys for other.
means T_BEGLEIT by age_cohort by gesl
\textit{cells = mean stddev count min max.}

*average time at home.
means home_time by age_cohort by gesl
  /cells = mean stddev count min max.

*average time out of home.
means out_time by age_cohort by gesl
  /cells = mean stddev count min max.

*average time out of home at activities.
means activity_time by age_cohort by gesl
  /cells = mean stddev count min max.