EPFL
ENAC TRANSP-OR
Prof. M. Bierlaire

Mathematical Modeling of Behavior
Fall 2018

Netherlands Mode Choice

The objective of the labs 10 and 11 is to become familiar with the usage of choice models in order to predict the aggregated market shares in some hypothetical situations, to compute relevant indicators for policy analysis and to forecast the prices that maximize the expected revenue. To this end, we will use the simulation instructions that are available in Biogeme.

With respect to the market shares, the idea is to predict them with a method called *sample enumeration*. In order to be able to use the Netherlands dataset for aggregation and prediction, we have to restrict it to contain only the revealed preference (RP) data (i.e., the actual choices). From now on, when we refer to the Netherlands dataset, we refer to its restricted version containing RP data only. This dataset is called `netherlandsRP.dat` and is provided.

In this exercise, you consider a choice model that has been estimated for the Netherlands case study (see `Netherlands_Base_Model.py`). The deterministic terms of the utility function are defined as follows:

$$\begin{aligned} V_{CAR} &= \mathrm{ASC}_{\mathrm{CAR}} + \beta_{\mathrm{Cost_Age1}} \cdot \mathrm{Cost}_{\mathrm{CAR}} \cdot \mathrm{Age}_1 + \beta_{\mathrm{Cost_Age2}} \cdot \mathrm{Cost}_{\mathrm{CAR}} \cdot \mathrm{Age}_2 \\ &\quad + \beta_{\mathrm{Time_CAR}} \cdot \mathrm{TravelTime}_{\mathrm{CAR}}, \\ V_{RAIL} &= \beta_{\mathrm{Cost_Age1}} \cdot \mathrm{Cost}_{\mathrm{RAIL}} \cdot \mathrm{Age}_1 + \beta_{\mathrm{Cost_Age2}} \cdot \mathrm{Cost}_{\mathrm{RAIL}} \cdot \mathrm{Age}_2 \\ &\quad + \beta_{\mathrm{Time_RAIL}} \cdot \mathrm{TravelTime}_{\mathrm{RAIL}} + \beta_{\mathrm{Female}} \cdot \mathrm{Female} + \beta_{\mathrm{ArrivalTime}} \cdot \mathrm{ArrivalTime}, \end{aligned}$$

where $\mathrm{Cost}_{\mathrm{CAR}}$ and $\mathrm{Cost}_{\mathrm{RAIL}}$ are the cost of car and rail in euros, respectively. $\mathrm{TravelTime}_{\mathrm{CAR}}$ and $\mathrm{TravelTime}_{\mathrm{RAIL}}$ are the total travel time of car and rail in hours, respectively. Age_1 is 1 if the individual is 40 years or younger, Age_2 is 1 if the individual is 41 years or older (i.e., $\mathrm{Age}_2 = 1 - \mathrm{Age}_1$), Female is 1 if the reported gender is female, and ArrivalTime is 1 if the individual must arrive at the destination by a given time.

1 Aggregation

Now, you may want to perform the aggregation to predict the market share. Assume that the procedure to collect the sample is *stratified random sampling*. Thus, it is necessary to associate a weight with each group or stratum. In our case, and given the available data and the relevant socioeconomic characteristics included in the choice model, we consider 4 strata:

1. Male and Age_1;
2. Male and Age_2;

3. Female and Age_1;

4. Female and Age_2.

In order to compute the weight associated with each group g, we need the total number of individuals in the population (N) and the number of individuals in the population belonging to each group (N_g). Even if the population should ideally contain only the individuals in the Netherlands commuting between Nijmegen and the Randstad in 1987 (scope of the case study), due to the unavailability of the data for this population, we consider the above mentioned strata for the whole country in 2011 as an instance.

	Age_1	Age_2
Male	4,092,390	4,151,092
Female	3,984,028	4,428,289

Table 1: Dutch Census 2011

Table 1 shows the size of the 4 groups in the whole population. Now, you need to identify the size of these four groups S_1, S_2, S_3 and S_4 within the sample of dataset you have. You can use a statistical software like Excel or R to do this. You should obtain the following results:

1. $S_1 = 89$,

2. $S_2 = 37$,

3. $S_3 = 65$, and

4. $S_4 = 37$.

Now, You need to calculate the weight associated with each individual. The weight of group g is then calculated as follows:

$$\omega_g = \frac{N_g}{N} \cdot \frac{S}{S_g}, \tag{1}$$

where $N = \sum_{g=1}^{4} N_g$ and $S = \sum_{g=1}^{4} S_g$. You should obtain the following weights:

1. $\omega_1 = 0.63$,

2. $\omega_2 = 1.54$,

3. $\omega_3 = 0.84$, and

4. $\omega_4 = 1.64$.

Finally, you need to associate a weight with each individual. As each individual n belongs to exactly one stratum g, the individual weight is defined as:

$$\omega_n = \sum_{g=1}^{4} \delta_{ng} \omega_g, \tag{2}$$

where δ_{ng} is equal to one if individual n belongs to stratum g and zero otherwise. Now, we just need to normalize the weights, i.e., $\sum_{n=1}^{S} \omega_n = 1$. We divide the individual weights by the sum of individual weights, i.e.,

$$\overline{\omega}_n = \omega_n \cdot \frac{1}{\sum_{n=1}^{N} \omega_n} = \frac{\omega_n}{S}. \tag{3}$$

The column containing the weights for each individual needs to be included in the Netherlands dataset (for instance, it can be called `Weights`) so that Biogeme can use this information for aggregation and prediction. You have to store this dataset and use it from now on.

Once the weights have been calculated and included in the dataset, you might want to exclude later on some observations satisfying some criteria, and therefore the weights will not sum up to 1 anymore. In this case, you need to renormalize the weights so that their sum is equal to 1. In general, in order to account for the weights in Biogeme, you should follow these steps:

1. **Renormalization**: to calculate the new sum of individual weights, the following instruction has to be included in the model file `Netherlands_Base_Model.py` (assume the column containing the weight is labeled as `Weights`):
   `BIOGEME_OBJECT.STATISTICS['Sum of weights'] = Sum(Weights,'obsIter').`
   In this way, the sum of weights will be shown on the `.html` file generated when running `Netherlands_Base_Model.py`. Note that, in this exercise, the sum should be already one since any observation is not excluded.

2. **Sample size**: Check the sample size on the generated file `Netherlands_Base_Model.html`. Note that it is not equivalent to the number of entries in the dataset if you exclude observations. In this exercise, the sample size remains unchanged since we are not excluding any observation.

3. **Weight in Biogeme**: According to the above information, a new variable (`theWeight`) needs to be defined in order to normalize the existing weights (`Weights`). The renormalized weights are calculated by multiplying the original weights (`Weights`) by the sample size (x) and dividing them by the total sum of weights (y), both obtained in the previous steps:
   ```
   theWeight = Weights * x/y
   BIOGEME_OBJECT.WEIGHT = theWeight
   ```
   In this exercise, you should obtain $x = 228$ and $y = 1$ and define the weights for Biogeme in this way.

2 Simulation file

Up to this point, you have added the column containing the weights to the Netherlands dataset. In order to construct the simulation file, you should follow the instructions provided in *Description of the simulation file*, which is available during the MOOC (`Course > 7. Forecasting > 7.1 Aggregation > Description of the simulation file`). You can call the new file for simulation `Netherlands_Base_Simul.py`. We ask you to use the `simulate` variable to conduct the following tasks:

1. *Compute the predicted market shares for car and rail with stratified random sampling.*
   The predicted market shares are obtained with the following instructions:
   `prob_CAR = bioLogit(V,av,0)`
   `prob_RAIL = bioLogit(V,av,1)`
   `simulate = {'Prob.  CAR': prob_CAR, 'Prob.  RAIL': prob_RAIL}`
   The aggregated market shares can be found in the row "Weighted average" from the resulting `.html` file:

   - car: 63.95%, and
   - rail: 36.05%.

2. *Compare the predicted market shares with the actual choices. More precisely calculate the following shares:*

   - *share of users choosing car with a higher probability for rail, and*
   - *share of users choosing rail with a higher probability for car.*

   Try to find the possible causes.
   You can easily drag the `.html` into a spreadsheet (e.g., Excel) to calculate the mentioned shares. You should obtain:

   - share of users choosing car with a higher probability for rail: 9.21% ,and
   - share of users choosing rail with a higher probability for car: 14.91%.

---

mpp / yo / tr