BIOGEME

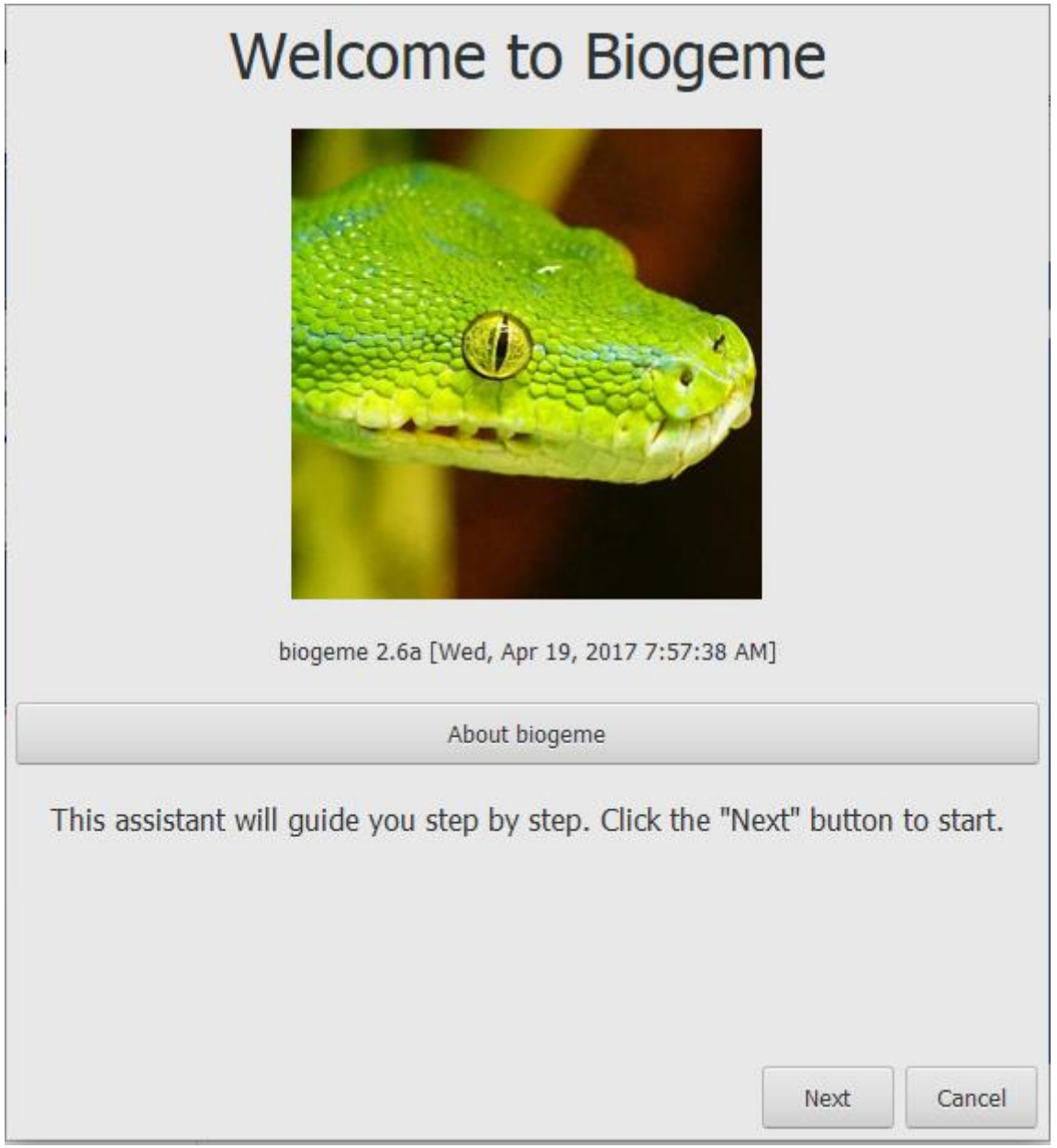
(Bierlaire's Optimization package for GEV Model Estimation)

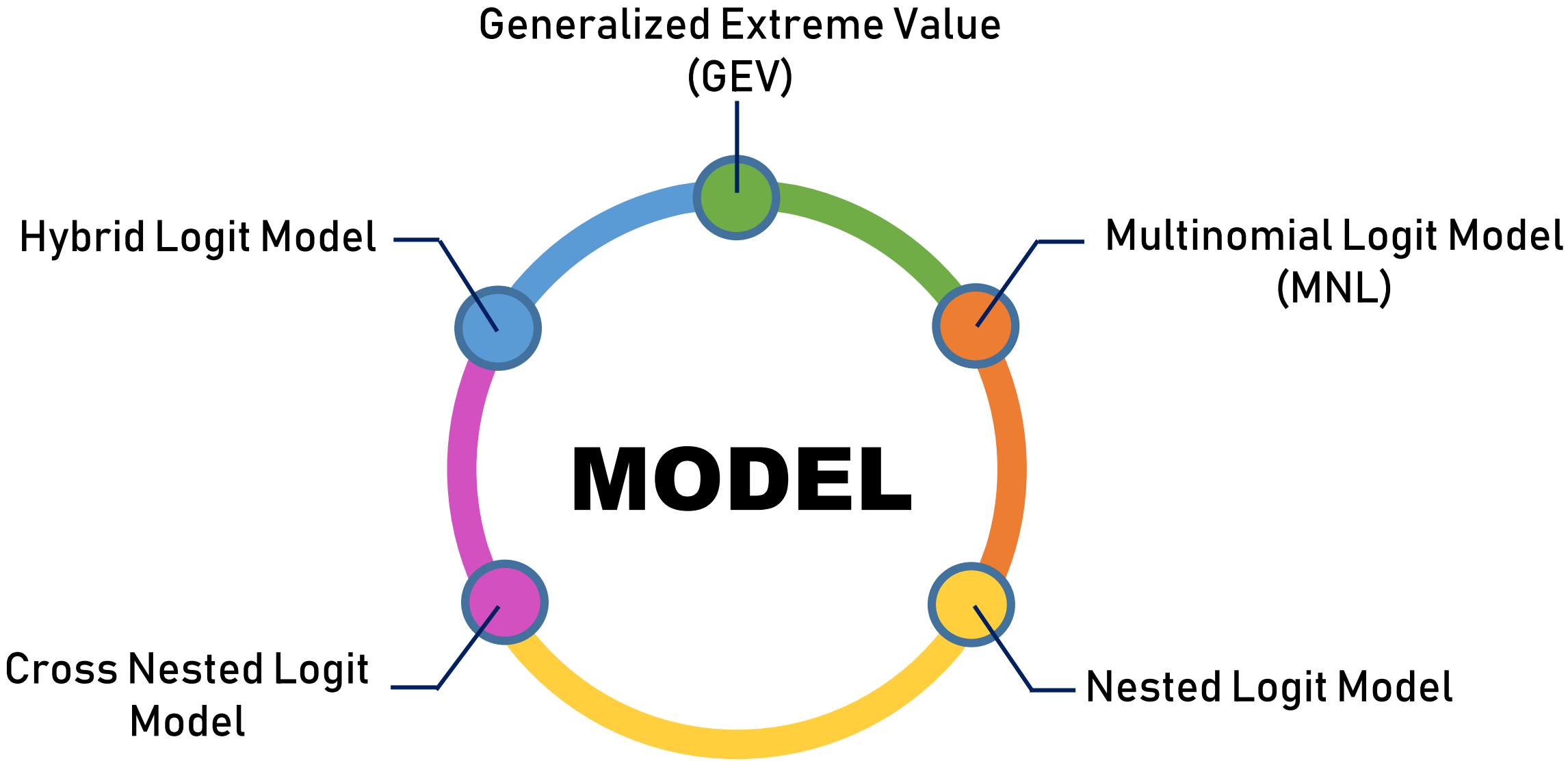
Sebuah *open source package* yang dirancang untuk memperkirakan berbagai utilitas bilangan acak yang berdasarkan nilai maximum likelihood.

- ❑ Pythonbiogeme
- ❑ Bisonbiogeme
- ❑ Biosim
- ❑ mod2py

Contoh penggunakan Biogeme :

- ➢ Perkiraan kebutuhan transportasi
- ➢ Pemodelan keputusan individual
  - Pemilihan moda transportasi
  - Pemilihan lokasi tempat tinggal
  - Pemilihan lokasi parkir
  - Pemilihan rencana perjalanan




Generalized Extreme Value (GEV)
Hybrid Logit Model
Multinomial Logit Model (MNL)
MODEL
Cross Nested Logit Model
Nested Logit Model

ELEMENT MODEL

1 Fixed parameter

Merupakan parameter pada model yang akan diestimasikan

B_TravelTime = Beta ('B_TravelTime', 0,-10,10,0)

2 Variables

Merupakan explanatory dari setiap variable yang terdapat dalam model

CAR_TravelTime_SC = DefineVariable ('CAR_TravelTime_SC', CAR_TravelTime / 10.0)

3 Random numbers

prob = bioLogit(V,av,CHOICE)

4 Mathematical expressions

Utility function
V = {1: CHOICE1, 2 :CHOICE2}

5 Iterator

Merupakan komponen dalam biogeme yang akan meakukan iterasi terhadap data masukan

Defines an iterator on the data :
rowIterator('obsIter')

6 Function Sum (term,iterator) dan Prod (term, iterator)

BIOGEME_OBJECT.ESTIMATE = Sum(log(prob),'obsIter')

KONSEP DASAR BIOGEME

Perancangan Model :

1. Utilitas masing- masing alternatif untuk setiap individu n
2. Probabilitas pilihan alternatif individu i
3. Dependent variable dinotasikan Y
4. Alternatif pilihan dinotasikan Yi
5. Independet variable dinotasikan X
6. Menentukan bentuk model (MNL, NL, CNL, Hybrid)
7. Spesifikasi utilitas pada masing-masing alternatif Ui

Contoh : Pemilihan Model pada Swissmetro

Spesifikasi model :

1. Pemilihan moda transportasi dengan 3 alternatif yaitu TRAIN, CAR, SWISSMETRO
2. Tipe survei data Stated Preference (SP) dan Revealed Preference (RP)
3. Variabel terkait yaitu Biaya Perjalanan, Waktu Tempuh, Jumlah frekuensi penggunaan transportasi umum, Usia responden, GA (Generalabomentent) tiket rail system, Kelas tempat duduk di Swissmetro, Pembawaaan bagasi

Spesifikasi model :

1. Pemilihan moda transportasi dengan 3 alternatif yaitu TRAIN, CAR, SWISSMETRO
2. Tipe survei data Stated Preference (SP) dan Revealed Preference (RP)
3. Variabel terkait yaitu Biaya Perjalanan, Waktu Tempuh, Jumlah frekuensi penggunaan transportasi umum, Usia responden, GA (Generalabomentent) tiket rail system, Pemesanan kursi di Swissmetro

1. Fungsi utilitas

$V1 = V_{TRAIN} = ASC_{TRAIN} + B_{TIME} * TRAIN\_TT\_SCALED + B_{COST} * TRAIN\_COST\_SCALED$
$V2 = V_{SM} = ASC_{SM} + B_{TIME} * SM\_TT\_SCALED + B_{COST} * SM\_COST\_SCALED$
$V3 = V_{CAR} = ASC_{CAR} + B_{TIME} * CAR\_TT\_SCALED + B_{COST} * CAR\_COST\_SCALED$

- TRAIN_TT_SCALED, TRAIN_COST_SCALED, SM_TT_SCALED, SM_COST_SCALED, CAR_TT_SCALED, CAR_COST_SCALED adalah variabel
- ASC_TRAIN, ASC_SM, ASC_CAR,B_TIME,B_COST adalah parameter yang akan diestimasi
- Probabilitas pemilihan alternatif yang tersedia (i) ditunjukkan dengan model logit :

$$P(i) = \exp(Vi) / (\exp(V1) + \exp(V2) + \exp(V3))$$

- Apabila dalam 1 set data sebanyak n-kali pengamatan, maka fungsi loglikelihood dari sampel :

$$L = \Sigma n \log P (in)$$

Dimana i_n adalah alternatif sebenarnya yang dipilih oleh individu n

2. Input Data

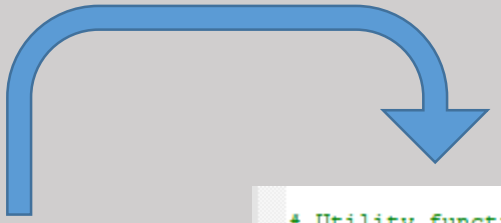
Biogeme mengasumsikan data file diisikan di baris pertama sebagai daftar tabel yang terkait dengan data yang tersedia dimana setiap baris berikutnya berisi data numerik yang sama persis jumlahnya dengan hasil observasi

ID	ASC_CAR	ASC_TRAIN	ASC_SM	B_TIME	B_COST	...
Obs1						
Obs2						
Obs3						
Obs4						

3. Spesifikasi Model

```
from biogeme import *
from headers import *
from loglikelihood import *
from statistics import *

#Parameters to be estimated
# Arguments:
#    - 1  Name for report; Typically, the same as the variable.
#    - 2  Starting value.
#    - 3  Lower bound.
#    - 4  Upper bound.
#    - 5  0: estimate the parameter, 1: keep it fixed.
#
ASC_CAR = Beta('Car cte.',0,-10,10,0)
ASC_TRAIN = Beta('Train cte.',0,-10,10,0)
ASC_SM = Beta('Swissmetro cte.',0,-10,10,1)
B_TIME = Beta('Travel time',0,-10,10,0)
B_COST = Beta('Travel cost',0,-10,10,0)
```



```
# Utility functions

#If the person has a GA (season ticket) her incremental cost is actually 0
#rather than the cost value gathered from the
# network data.

SM_COST =  SM_CO   * (  GA   ==  0  )
TRAIN_COST =  TRAIN_CO   * (  GA   ==  0  )

# For numerical reasons, it is good practice to scale the data to
# that the values of the parameters are around 1.0.
# A previous estimation with the unscaled data has generated
# parameters around -0.01 for both cost and time. Therefore, time and
# cost are multipled my 0.01.

# The following statements are designed to preprocess the data. It is
# like creating a new columns in the data file. This should be
# preferred to the statement like
# TRAIN_TT_SCALED = TRAIN_TT / 100.0
# which will cause the division to be reevaluated again and again,
# throuh the iterations. For models taking a long time to estimate, it
# may make a significant difference.

TRAIN_TT_SCALED = DefineVariable('TRAIN_TT_SCALED', TRAIN_TT / 100.0)
TRAIN_COST_SCALED = DefineVariable('TRAIN_COST_SCALED', TRAIN_COST / 100)
SM_TT_SCALED = DefineVariable('SM_TT_SCALED', SM_TT / 100.0)
SM_COST_SCALED = DefineVariable('SM_COST_SCALED', SM_COST / 100)
CAR_TT_SCALED = DefineVariable('CAR_TT_SCALED', CAR_TT / 100)
CAR_CO_SCALED = DefineVariable('CAR_CO_SCALED', CAR_CO / 100)

V1 = ASC_TRAIN + B_TIME * TRAIN_TT_SCALED + B_COST * TRAIN_COST_SCALED
V2 = ASC_SM + B_TIME * SM_TT_SCALED + B_COST * SM_COST_SCALED
V3 = ASC_CAR + B_TIME * CAR_TT_SCALED + B_COST * CAR_CO_SCALED
```

```
# Associate utility functions with the numbering of alternatives
V = {1: V1,
     2: V2,
     3: V3}

# Associate the availability conditions with the alternatives
CAR_AV_SP =  DefineVariable('CAR_AV_SP',CAR_AV  * (  SP   !=  0  ))
TRAIN_AV_SP =  DefineVariable('TRAIN_AV_SP',TRAIN_AV  * (  SP   !=  0  ))

av = {1: TRAIN_AV_SP,
      2: SM_AV,
      3: CAR_AV_SP}

# The choice model is a logit, with availability conditions
prob = bioLogit(V,av,CHOICE)

# Defines an itertor on the data
rowIterator('obsIter')

# DEfine the likelihood function for the estimation
BIOGEME_OBJECT.ESTIMATE = Sum(log(prob),'obsIter')

# All observations verifying the following expression will not be
# considered for estimation
# The modeler here has developed the model only for work trips.
# Observations such that the dependent variable CHOICE is 0 are also removed.
exclude = (( PURPOSE != 1 ) * (  PURPOSE   !=  3  ) + ( CHOICE == 0 )) > 0

BIOGEME_OBJECT.EXCLUDE = exclude

# Statistics

nullLoglikelihood(av,'obsIter')
choiceSet = [1,2,3]
cteLoglikelihood(choiceSet,CHOICE,'obsIter')
availabilityStatistics(av,'obsIter')
```

5. Output Biogeme

Statistics

Alt. 1 available:	*6768*
Alt. 1 chosen:	*908*
Alt. 2 available:	*6768*
Alt. 2 chosen:	*4090*
Alt. 3 available:	*5607*
Alt. 3 chosen:	*1770*
Cte loglikelihood (only for full choice sets):	*-6257.86*
Null loglikelihood:	*-6964.66*

Estimation report

Number of estimated parameters:	4
Sample size:	6768
Excluded observations:	3960
Init log likelihood:	-6964.663
Final log likelihood:	-5331.252
Likelihood ratio test for the init. model:	3266.822
Rho-square for the init. model:	0.235
Rho-square-bar for the init. model:	0.234
Akaike Information Criterion:	10670.504
Bayesian Information Criterion:	10697.784
Final gradient norm:	+6.288e-004
Diagnostic:	Trust region algorithm with simple bounds
Iterations:	4
Data processing time:	00:00
Run time:	00:00
Nbr of threads:	2

Estimated parameters

Click on the headers of the columns to sort the table [Credits]

Name	Value	Std err	t-test	p-value	Robust Std err	Robust t-test	p-value
Car cte.	-0.155	0.0432	-3.58	0.00	0.0582	-2.66	0.01
Train cte.	-0.701	0.0549	-12.78	0.00	0.0826	-8.49	0.00
Travel cost	-1.08	0.0518	-20.91	0.00	0.0682	-15.89	0.00
Travel time	-1.28	0.0569	-22.46	0.00	0.104	-12.26	0.00

Correlation of coefficients

Click on the headers of the columns to sort the table [Credits]

Coefficient1	Coefficient2	Covariance	Correlation	t-test	p-value	Rob. cov.	Rob. corr.	Rob. t-test	p-value	
Travel cost	Travel time	0.000550	0.187	2.79	0.01	0.00220	0.309	1.84	0.07	*
Train cte.	Travel time	-0.00225	-0.722	5.56	0.00	-0.00760	-0.883	3.18	0.00	
Train cte.	Travel cost	8.22e-006	0.00289	5.08	0.00	-0.000831	-0.147	3.34	0.00	
Car cte.	Travel time	-0.00144	-0.585	12.57	0.00	-0.00482	-0.796	7.27	0.00	
Car cte.	Travel cost	0.000485	0.216	15.52	0.00	2.86e-005	0.00722	10.40	0.00	
Car cte.	Train cte.	0.00138	0.580	11.85	0.00	0.00390	0.812	11.16	0.00	

Smallest singular value: 5.90497

Reference

- http://biogeme.epfl.ch/home.html
- http://biogeme.epfl.ch/install.html

- http://biogeme.epfl.ch/examples_swissmetro.html
- http://biogeme.epfl.ch/examples_montecarlo.html
- http://biogeme.epfl.ch/examples_indicators.html
- http://biogeme.epfl.ch/examples_latent.html