

基礎ゼミ6

PythonBiogemeによる
NL, MXLのパラメータ推定
2018/5/18

福田研究室
M1 河井智弘

本日の内容

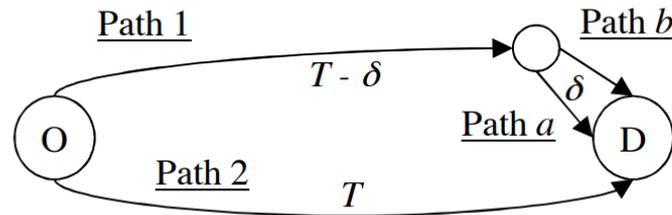
- Nested Logitモデルの概要
- Pythonbiogemeを用いてNested Logitモデルの演習
～NL_feedback.py, YokohamaPT_Simplified.dat～
- Mixed Logitモデルの概要
- Pythonbiogemeを用いたMixed Logitモデルの演習
～MXL_feedback.py, YokohamaPT_Simplified.dat～

IIA特性の問題点を緩和したい

☆ 選択肢のうち一部に生じた相関を何とか表現したい
⇒ Nested logit (NL) Model, Mixed Logit Model (MXL)を使う

※ 選択肢間の誤差項に相関が生じやすい例

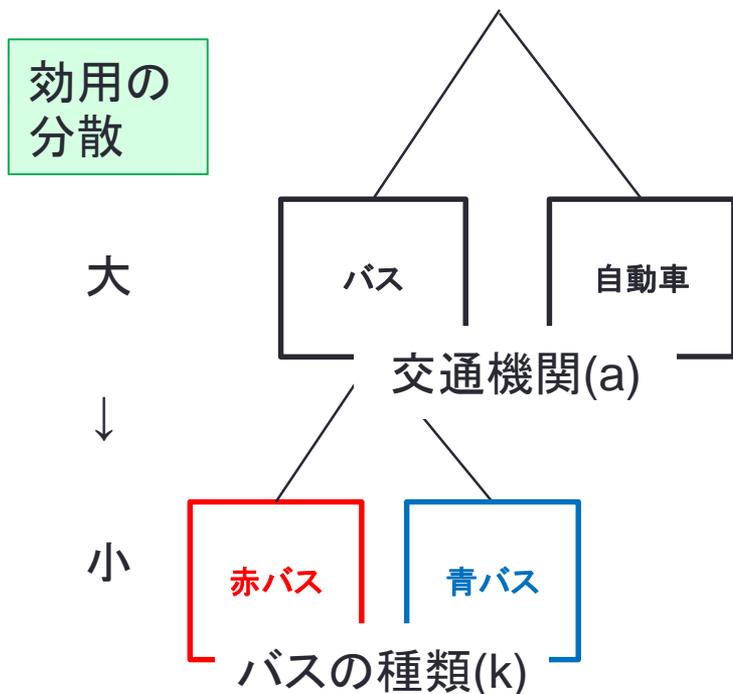
- 交通手段選択(赤バス, 青バス)
- 経路選択(一部のリンクを共有している経路間には相関が生じやすい)



- 目的地選択(近い目的地どうしには相関が生じやすい)

NL(ツリー構造の説明)

ツリー構造;最初に属性ごとに分ける
⇒条件付確率の考え方



$$V_a = \frac{1}{\mu_a} \ln \left(\sum_{m \in M} \exp(\mu_m V_m) \right)$$

ログサム変数

ネスト構造(この式では上位ネスト)から
得られる効用の最大値の期待値

$$\begin{aligned} P(a, k) &= P(k|a)P(a) \\ &= \frac{\exp(\mu_k V_k)}{\sum_i \exp(\mu_k V_i)} \times \frac{\exp\{\mu_a (V_a + V_a)\}}{\sum_j \exp\{\mu_a (V_j + V_j)\}} \end{aligned}$$

NL(ツリー構造と効用式)

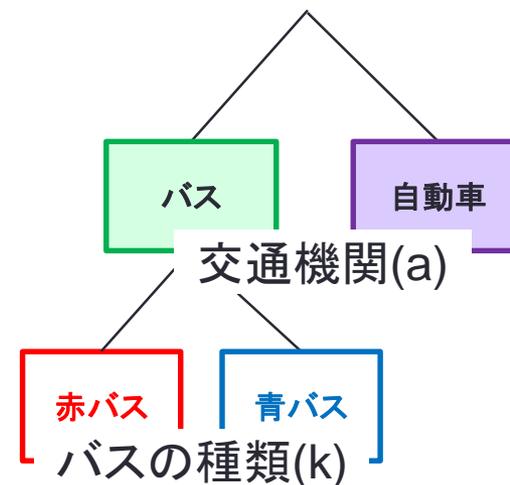
$$U_{Redbus} = \beta X_{Redbus} + \varepsilon_{bus} + \varepsilon'_{Redbus}$$

$$U_{Bluebus} = \beta X_{Bluebus} + \varepsilon_{bus} + \varepsilon'_{Bluebus}$$

$$U_{Car} = \beta X_{Car} + \varepsilon_{Car}$$

誤差項 ε

(IIDガンベル分布に従う)

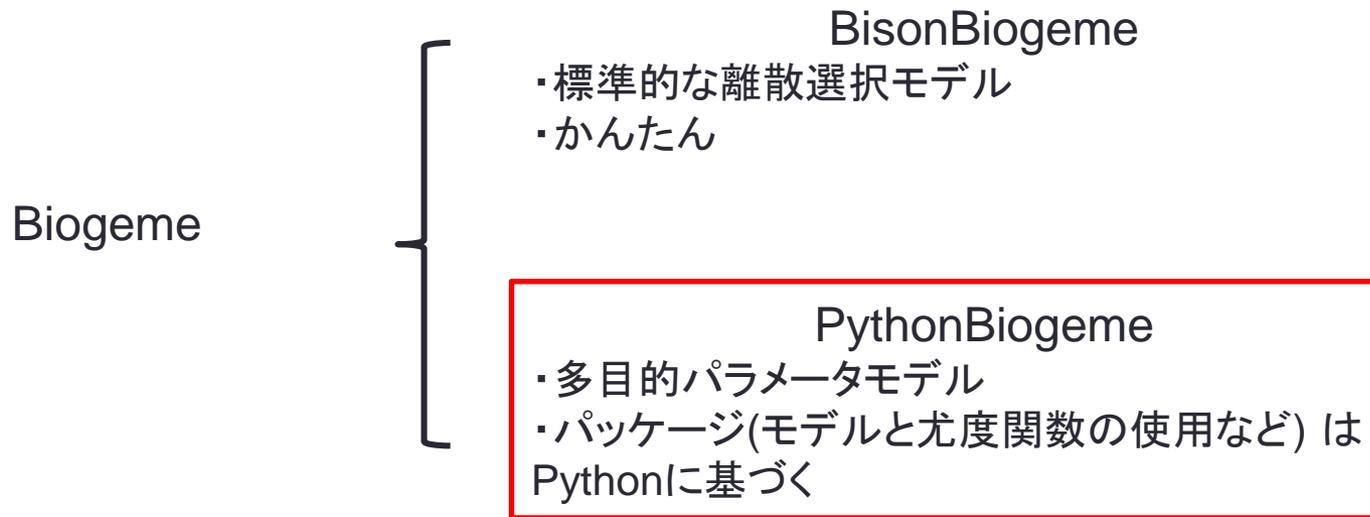


分散共分散行列 Σ は、

$$\Sigma = cov(U) = \begin{pmatrix} 0 & \sigma_{bus}^2 & 0 \\ \sigma_{bus}^2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} \sigma^2 & 0 & 0 \\ 0 & \sigma^2 & 0 \\ 0 & 0 & \sigma^2 \end{pmatrix}$$

$$\varepsilon = \varepsilon_{bus} + \varepsilon'$$

Pythonbiogemeについて



●Pythonとの類似点

#の後はコメント文として認識され, コードには影響を及ぼさない

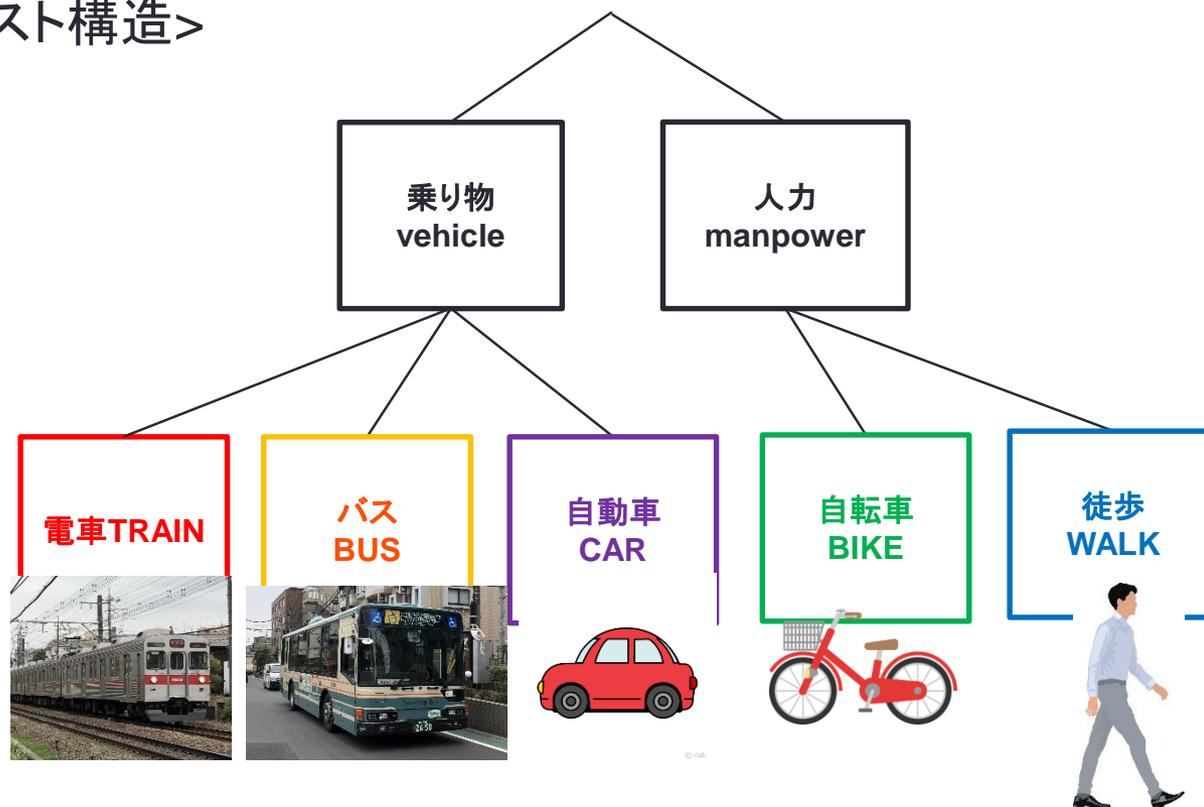
●Biogemeとの類似点

データは.datファイル
コードは.pyファイル } に変換して解析にかける

今回使用するデータ

- 前回と同じく横浜市内の交通移動データ; YokohamaPT_Full_Simplified.dat

<今回のネスト構造>



NL(コード導入で実践)

●NL_feedback.py ~ライブラリの読み込み~

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

必須のライブラリ

- ・biogeme...pythonbiogeme に必要なpython言語の拡張
- ・headers...dataの先頭行の読み込み

追加のライブラリ

- ・nested...Nested Logit 用のライブラリ
- ・loglikelihood...最尤推定に必要なライブラリ
- ・statistics...統計に使う式のライブラリ

NL:コードで実践

●NL_feedback.py ~パラメータの設定~

例: `ASC_TRAIN=Beta('ASC_TRAIN', 0, -10, 10, 0)`

```
ASC_TRAIN = Beta('ASC_TRAIN',0,-10,10,0)
ASC_BUS = Beta('ASC_BUS',0,-10,10,0)
ASC_CAR = Beta('ASC_CAR',0,-10,10,0)
ASC_WALK = Beta('ASC_WALK',0,-10,10,1)
ASC_BIKE = Beta('ASC_BIKE',0,-10,10,0)
B_TIME = Beta('B_TIME',0,-10,10,0)
B_COST = Beta('B_COST',0,-10,10,0)
```

パラメータ名

初期値

上限・下限

固定(1) or
非固定(0)

固定
(fixed)

```
MU = Beta('MU',1,1,10,0)
```

μ ; スケールパラメータ(Nested Logitモデルにおける下位ネストの相関を表す)

NL:コードで実践

- NL_feedback.py ～ネスト構造以外の効用式の設定～

```
V1 = ASC_TRAIN + B_TIME * TRAIN_TT_SCALED + B_COST * TRAIN_CO_SCALED
V2 = ASC_BUS + B_TIME * BUS_TT_SCALED + B_COST * BUS_CO_SCALED
V3 = ASC_CAR + B_TIME * CAR_TT_SCALED
V4 = ASC_WALK + B_TIME * WALK_TT_SCALED
V5 = ASC_BIKE + B_TIME * BIKE_TT_SCALED
```

NL:コードで実践

●NL_feedback.py ～スケール調整～

```
TRAIN_TT_SCALED = DefineVariable('TRAIN_TT_SCALED', TotalTimeTrain / 100)
TRAIN_CO_SCALED = DefineVariable('TRAIN_CO_SCALED', FareTrain / 100)
BUS_TT_SCALED = DefineVariable('BUS_TT_SCALED', TotalTimeBus / 100)
BUS_CO_SCALED = DefineVariable('BUS_CO_SCALED', FareBus / 100)
CAR_TT_SCALED = DefineVariable('CAR_TT_SCALED', TimeCar / 100)
WALK_TT_SCALED = DefineVariable('WALK_TT_SCALED', TimeWalk / 100)
BIKE_TT_SCALED = DefineVariable('BIKE_TT_SCALED', TimeBike / 100)
```

TRAIN_TT_SCALED

= DefineVariable('TRAIN_TT_SCALED', TotalTimeTrain / 100)

↓
変数定義の関数

↓
変数名

↓
スケール調整

TT; Travel Time(旅行時間)

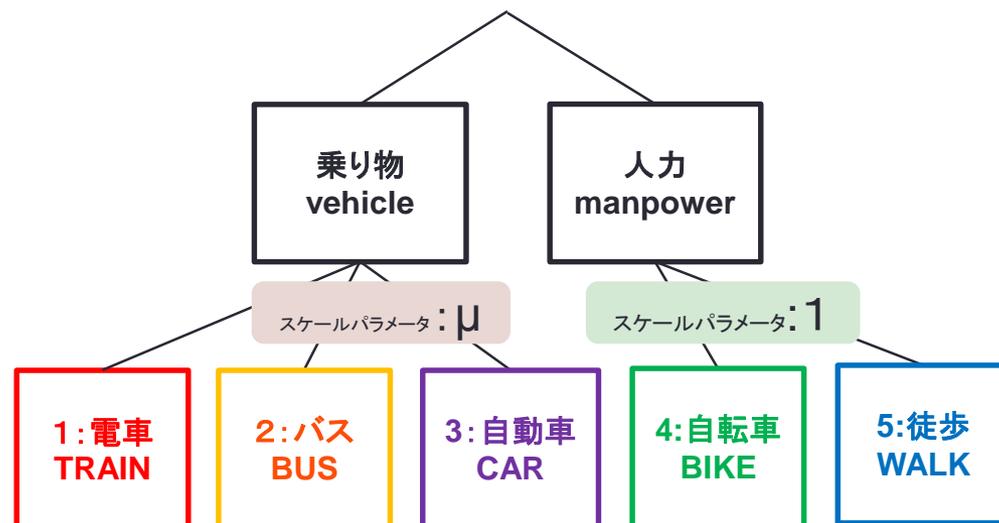
CO; Cost(費用; 本データでは公共交通のみ)

NL:コードで実践

●NL_feedback.py ~ネスト構造の設定~

```
# Definition of nests:
# 1: nests parameter
# 2: lists of alternatives
vehicle = MU, [1,2,3]
manpower = 1.0, [4,5]
nests = vehicle,manpower
```

[下位ネストのスケールパラメータ]
manpowerを1とし、
vehicleを μ とする



NL:コードで実践

●NL_feedback.py ～ネスト構造の設定～

```
# The choice model is a nested logit, with availability conditions  
logprob = lognested(V,av,nests,Choice)
```

Nestedモデル選択確率の対数; lognested(効用関数, availability, nests, choice)

```
# Defines an iterator on the data  
rowIterator('obsIter')
```

イテレータの設定

```
# Define the likelihood function for the estimation  
BIOGEME_OBJECT.ESTIMATE = Sum(logprob,'obsIter')
```

推定変数の設定

※av: 利用可能条件を表す

```
# Associate the availability conditions with the alternatives  
av = {1: ModeAvailableTrain,  
      2: ModeAvailableBus,  
      3: ModeAvailableCar,  
      4: ModeAvailableWalk,  
      5: ModeAvailableBike}
```

NL:コードで実践

●NL_feedback.py ～統計量の算出～

```

① nullLoglikelihood(av, 'obsIter')
   choiceSet = [1,2,3,4,5]
② cteLoglikelihood(choiceSet, Choice, 'obsIter')
③ availabilityStatistics(av, 'obsIter')

```

- ① nullLoglikelihood(利用可能な選択肢ID, データのイテレータ)
⇒ヌルログの計算

$$L(0) = - \sum_n \ln(J_n)$$

- ② cteLoglikelihood(選択肢, 実際の選択, データのイテレータ)
⇒パラメータが1つ固定の場合の対数尤度

$$L = \sum_n n_i \ln(n_i) - n \ln(n)$$

- ③ availabilityStatistics(availability, データのイテレータ)
⇒選択肢が利用可能である回数の統計値

結果; NL_feedback.html

Statistics

```

Alt. 1 available: 1268
  Alt. 1 chosen: 524
Alt. 2 available: 384
  Alt. 2 chosen: 41
Alt. 3 available: 1516
  Alt. 3 chosen: 512
Alt. 4 available: 1516
  Alt. 4 chosen: 231
Alt. 5 available: 1516
  Alt. 5 chosen: 208
Cte loglikelihood (only for full choice sets): -2108.22
Null loglikelihood: -2126.61

```

パラメーター1つ固定の対数尤度
L(0)

Estimation report

```

Number of estimated parameters: 7
  Sample size: 1516
Excluded observations: 0
  Init log likelihood: -2126.687
  Final log likelihood: -1342.607
Likelihood ratio test for the init. model: 1568.161
  Rho-square for the init. model: 0.369
Rho-square-bar for the init. model: 0.365
  Akaike Information Criterion: 2699.214
  Bayesian Information Criterion: 2736.481
  Final gradient norm: +3.428e-003
  Diagnostic: Trust region algorithm with simple bounds (CGT2000): Convergence reached...
  Iterations: 15
Data processing time: 00:00
  Run time: 00:01
Nbr of threads: 2

```

尤度比 ρ^2 など

結果; NL_feedback.html

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	
ASC_BIKE	-1.03	0.106	-9.69	0.00		0.152	-6.73	0.00	
ASC_BUS	-0.964	0.177	-5.45	0.00		0.339	-2.84	0.00	
ASC_CAR	-0.887	0.134	-6.64	0.00		0.293	-3.03	0.00	
ASC_TRAIN	0.561	0.116	4.82	0.00		0.115	4.87	0.00	
B_COST	-0.000541	0.0197	-0.03	0.98	*	0.0273	-0.02	0.98	*
B_TIME	-7.25	0.535	-13.55	0.00		1.67	-4.35	0.00	
MU	1.62	0.153	10.57	0.00		0.373	4.33	0.00	

パラメタ
推定値

標準偏差
 σ_k

t値
 t_k

p値

ロバスト推定による
推定値

ASC_WALK(fixed)
に対する値

※ロバスト推定
;外れ値を取り除いて
パラメータ値を推定

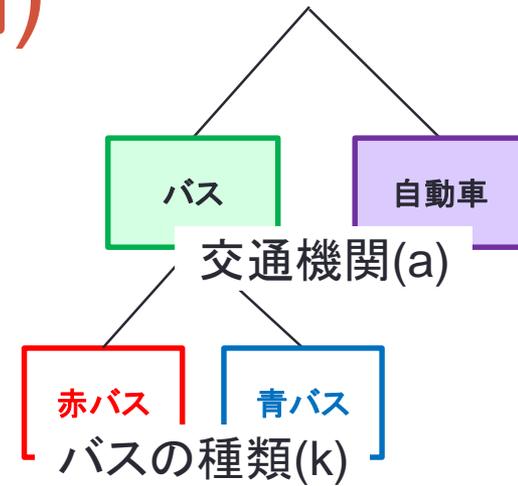
MXLの概要(赤バス青バス例)

$$\begin{aligned}
 U_{Redbus} &= \beta X_{Redbus} + \sigma_{bus} \tau_{bus} + \varepsilon'_{Redbus} \\
 U_{Bluebus} &= \beta X_{Bluebus} + \sigma_{bus} \tau_{bus} + \varepsilon'_{Bluebus} \\
 U_{Car} &= \beta X_{Car} + \varepsilon'_{Car}
 \end{aligned}$$

正規分布に従う(プロビットに類似)

誤差項 ε

IIDガンベル分布
(通常のロジットと同様)



分散共分散行列 Σ は、

$$\Sigma = cov(U) = \begin{pmatrix} \sigma_{bus}^2 & \sigma_{bus}^2 & 0 \\ \sigma_{bus}^2 & \sigma_{bus}^2 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} \sigma^2 & 0 & 0 \\ 0 & \sigma^2 & 0 \\ 0 & 0 & \sigma^2 \end{pmatrix}$$

$$\varepsilon = \varepsilon_{bus} + \varepsilon'$$

NL/MXLで誤差構造を扱う意義

●誤差要素モデル; 選択肢間の誤差相関を考慮できる
(IIA特性; 赤バス青バス, 経路重複問題など)

これまで

●ランダム係数モデル; パラメータ β の確率変動を考慮できる
(同じ値段でも人によって感じ方が違う;
学生は貧乏だから落ちた100円玉拾うけど
ビルゲイツは100円玉拾う時間をもったいない
⇒人によって効用への影響度も異なる など...)

ここから

ランダム係数モデル

- 効用関数のパラメータ β が個人 n によって変化する場合
効用関数

$$\begin{aligned}
 U_{in} &= \beta'_n X_{in} + \epsilon_{in} \\
 &= (\bar{\beta} + \tau_n)' X_{in} + \epsilon_{in} \\
 &= \bar{\beta}' X_{in} + \tau'_n X_{in} + \epsilon_{in} \\
 &= \bar{\beta}' X_{in} + \eta_{in} + \epsilon_{in}
 \end{aligned}$$

MXLモデルの形式になっている！

$$[\beta_n = \bar{\beta} + \tau_n]$$

パラメータのバラツキ(正規分布)を表す

選択確率

$$\begin{aligned}
 P_i &= \int \frac{\exp(\beta'_n X_{in})}{\sum_j \exp(\beta'_n X_{jn})} f(\beta_n | \Omega_\beta) d\beta_n \\
 &= \int \frac{\exp(\bar{\beta}' x_{in} + \tau'_n x_{in})}{\sum_j \exp(\bar{\beta}' x_{jn} + \tau'_n x_{jn})} f(\tau_n | \Omega_\tau) d\tau_n
 \end{aligned}$$

シミュレーション法(モンテカルロ法)の一例

$$P_{in} = \int L_{in}(\eta) f(\eta|\Omega) d\eta, \quad L_{in} = \frac{\exp(\beta' X_{in} + \eta_{in})}{\sum_j \exp(\beta' X_{in} + \eta_{in})}$$

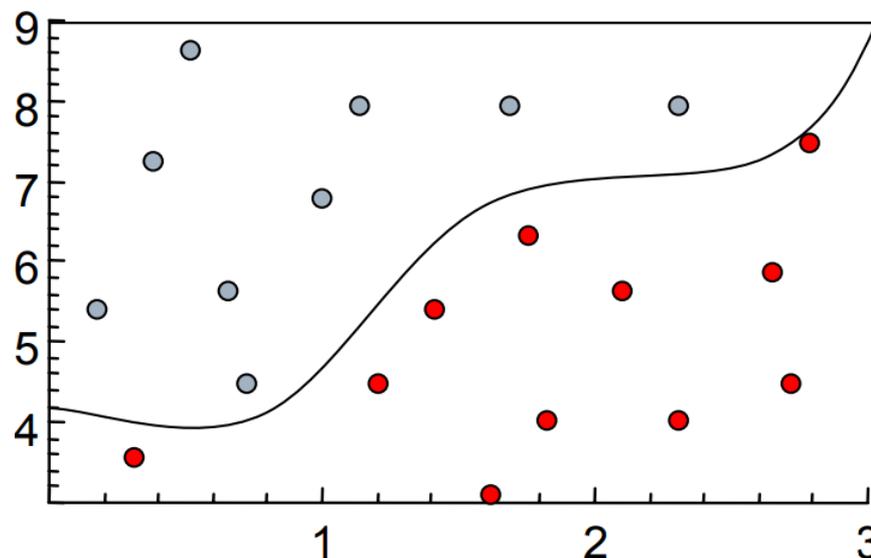
上記の積分を面積を用いて求める.

●2次元乱数を[3×9]の範囲に散らばらせる
⇒積分したい関数より上(青丸)or下(赤丸)を判定

●点の合計数は20
(←青9個, 赤11個)

⇒関数の積分結果は

$$3 \times 9 \times 11/20 = 14.85$$



MXL:コードで実践

- MXL_feedback.py～シミュレーション積分～
ランダム係数モデル(所要時間のパラメータB_TIMEが変化)

```
# Define a random parameter, log normally distributed, designed to be used  
# for Monte-Carlo simulation  
B_TIME_RND = -exp(B_TIME + B_TIME_S * bioDraws('B_TIME_RND'))
```

↓
繰り返し計算

↓
 $\sigma_\beta; \beta$ の標準偏差

↓
シミュレーション積分

MXL:コードで実践

●MXL_feedback.py ～プロビットモデルの挿入～

```

BIOGEME_OBJECT.PARAMETERS['optimizationAlgorithm'] = "BIO"
BIOGEME_OBJECT.DRAWS = { 'B_TIME_RND': 'NORMAL' }
BIOGEME_OBJECT.FORMULAS['Train utility'] = V1
BIOGEME_OBJECT.FORMULAS['Bus utility'] = V2
BIOGEME_OBJECT.FORMULAS['Car utility'] = V3
BIOGEME_OBJECT.FORMULAS['Walk utility'] = V4
BIOGEME_OBJECT.FORMULAS['Bike utility'] = V5

```

正規分布に従うランダム係数項
(プロビットモデルに類似)

$$U_{in} = \bar{\beta}' X_{in} + \tau'_n X_{in} + \epsilon_{in}$$

正規分布
に従う

IIDガンベル
分布に従う

MXL:コードの結果

Cte loglikelihood (only for full choice sets): -2118.17
 Null loglikelihood: -2135.67

Estimation report

```

Number of draws: 500
Number of estimated parameters: 7
Sample size: 1522
Excluded observations: 0
Init log likelihood: -1786.485
Final log likelihood: -1286.226
Likelihood ratio test for the init.
model: 1000.516
Rho-square for the init. model: 0.280
Rho-square-bar for the init. model: 0.276
Akaike Information Criterion: 2586.453
Bayesian Information Criterion: 2623.747
Final gradient norm: +4.497e-003
  
```

Estimated parameters

Click on the headers of the columns to sort the table

Name	Value	Std err	t-test	p-value	
ASC_BIKE	-1.82	0.147	-12.43	0.00	
ASC_BUS	-1.74	0.206	-8.43	0.00	
ASC_CAR	-2.07	0.156	-13.31	0.00	
ASC_TRAIN	0.512	0.167	3.07	0.00	
B_COST	-0.00144	0.0370	-0.04	0.97	*
B_TIME	2.69	0.0706	38.05	0.00	
B_TIME_S	0.737	0.0633	11.65	0.00	

参考文献

- 技報堂出版「交通行動の分析とモデリング」
- 羽藤研 理論勉強会第4回 <http://bin.t.u-tokyo.ac.jp/startup14/file/2-3.pdf>
- 羽藤研 BinN studiesシリーズ <http://bin.t.u-tokyo.ac.jp/kaken/>
- 夏の学校2008「効用最大化に基づく離散選択モデルの基礎(暫定版)」 <http://bin.t.u-tokyo.ac.jp/model08/lecture/kurauchi.pdf>
- 「パラメータ推定の実践的テクニック」
<http://bin.t.u-tokyo.ac.jp/model08/lecture/sasaki.pdf>
- 福田先生資料(OCW-i)
- Biogeme公式サイト(スイスmetroのサンプルコードを参考に)
http://biogeme.epfl.ch/examples_swissmetro.html