動態マクロ経済学

Week 3

佐藤健治
sato@eco.osakafu-u.ac.jp

2020/5/22

準備運動：IPython を起動してください

- Python には**リスト**という数学でいうところのベクトルっぽいオブジェクトがある。

```
x = [1, 2, 3]
y = [0.1, 0.2, 0.3]
```

- でもベクトルとしては使えない。

```
x + y
```

```
[1, 2, 3, 0.1, 0.2, 0.3]
```

NumPy

- Python の数値計算は NumPy を使うのが事実上の標準になっている。
- 次のコードを IPython 起動のたびに（Jupyter ならノートごとに）実行する。

```
import numpy as np
```

- np. 関数名 () という書き方で　NumPy の関数を使える。

```
np.log(np.exp(1.0))
```

```
1.0
```

ベクトル

- np.array() という関数に，Python のリストを渡せば「ベクトル」を表現するオブジェクトができる。

```
x = np.array([1, 2, 3])
y = np.array([0.1, 0.2, 0.3])
x + y
```

```
array([1.1, 2.2, 3.3])
```

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \ \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \Rightarrow \boldsymbol{x} + \boldsymbol{y} = \begin{bmatrix} 1.1 \\ 2.2 \\ 3.3 \end{bmatrix}$$

内積

- 内積：同じ長さ（N としよう）のベクトル 2 つに対して

$$\boldsymbol{x} \cdot \boldsymbol{y} = x_1 y_1 + \cdots + x_N y_N = \sum_{n=1}^{N} x_n y_n$$

  1. 要素ごとの積を計算して，$(\boldsymbol{x}, \boldsymbol{y}) \mapsto (x_1 y_1, \cdots, x_N y_N)$
  2. 総和を取る。$(x_1 y_1, \cdots, x_N y_N) \mapsto \sum_{n=1}^{N} x_n y_n$

- NumPy では `*` で要素ごとの積を計算できる。総和は `np.sum()`

```
np.sum(x * y)
```

```
1.4
```

- `np.dot(x, y)` としてもよい（その方が効率的）。

本日の目標

- テキスト 第 2 章
  - 2 週間かかると言ってましたがそれは 3 章の話でした。すいません。
- やること
  - ベクトルの計算（基本の説明は終わり）
  - 価格指数の定義
  - 価格指数の計算

ベクトルの内積がたくさん出てくるので，さっきの Python コードを意識しながら聞いてください。

指数化と指数の変化

指数化

- 先週は単一の変数の変化率を定義した。
- 今週の目標は、
  - 複数の変数を合成して単一の変数（指数）に変換する方法
  - その指数の変化

価格ベクトル

経済にはたくさんの財・サービスがあるので，価格も様々。
価格を並べた「ベクトル」として表現。

$$\boldsymbol{p}_t = \begin{bmatrix} p_{t,1} \\ \vdots \\ p_{t,n} \\ \vdots \\ p_{t,N} \end{bmatrix}$$

t は期を表す変数，$n = 1, \ldots, N$ は財のインデックス

価格の変化

価格の変化をどう表現する？

$$\boldsymbol{p}_{t-1} = \begin{bmatrix} p_{t-1,1} \\ \vdots \\ p_{t-1,n} \\ \vdots \\ p_{t-1,N} \end{bmatrix} \longrightarrow \boldsymbol{p}_t = \begin{bmatrix} p_{t,1} \\ \vdots \\ p_{t,n} \\ \vdots \\ p_{t,N} \end{bmatrix}$$

個々の財価格の上がり方ではなく，経済全体の平均的な価格上昇について知りたいとしよう。

価格指数

- 価格ベクトルを一定のルールのもとに単一の数に変換したもの。
  - 基準時点（$t = 0$）の指数を 1 とする。(100 にすることが多いけど面倒だから 1)

$$\begin{array}{ccccc} \boldsymbol{p}_0 & \to & \boldsymbol{p}_{t-1} & \to & \boldsymbol{p}_t \\ \updownarrow & & \updownarrow & & \updownarrow \\ P_0 = 1 & \to & P_{t-1} & \to & P_t \end{array}$$

- 変化率が重要 (1 に基準化してるので値に意味はない)。

$$\frac{\Delta P_t}{P_{t-1}} = \frac{P_t}{P_{t-1}} - 1 = \text{インフレ率 (inflation rate)}$$

用語：インフレ・デフレと物価の安定

- **インフレーション**＝物価指数の継続的な上昇。インフレ
  - 激しいインフレを正常化する政策介入→「**ディスインフレーション**」
- **デフレーション**＝物価指数の継続的な下落。デフレ
  - デフレになると「将来，価格が下がるかもしれないから，今は買わないでおこう」と考える人が増えて不況が長引く。
  - デフレーションの状態にある経済で超緩和的な政策をしてインフレを引き起こそうという政策介入は「リフレーション」
- 中央銀行が目標にしている「**物価の安定**」というのは，適度なインフレ状態（2% 程度がよいと言われている）が継続する状態
  - 賃金も徐々に上がるはずなので，生計が悪化する訳ではない
  - 長期では貨幣供給量の上昇率と同率で物価が上昇すると考えられていて，日銀は貨幣供給量を 2% 程度で増やしている（k % ルール）

価格指数を発見する

「価格指数は変化率を表現している」ということが分かれば，定義を覚えるのも難しくない。「発見的」に考えていこう。

- P_t/P_{t-1} を知りたいので，

$$\frac{\boldsymbol{p}_t}{\boldsymbol{p}_{t-1}}$$

  みたいなものを計算したい。でもベクトルの割算はできない。

- 内積をとって数にすれば割算ができる。（ドットは「内積」）

$$\frac{\boldsymbol{p}_t \cdot \square}{\boldsymbol{p}_{t-1} \cdot \square}$$

  この箱に何を入れるか？ を考える。

価格指数を発見する（続き）

- 経済活動の規模を反映したものにしたいので，取引量 $\boldsymbol{x}_{t-1}$, $\boldsymbol{x}_t$ を使えば良さそう。でも，単純に

$$\frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_t}{\boldsymbol{p}_t \cdot \boldsymbol{x}_{t-1}}$$

  とすると，価格変化と数量変化の両方が影響してしまうから不適。
- 取引量はどこかの時点で固定しないといけなさそう。これで完成。

$$\frac{P_t}{P_{t-1}} = \frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_{t-1}}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_{t-1}} \qquad \text{または} \qquad \frac{P_t}{P_{t-1}} = \frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_t}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_t}$$

  - 前者が**ラスパイレス式価格指数（連鎖方式）**
  - 後者が**パーシェ式価格指数（連鎖方式）**

指数の再帰的な計算

- 先程の公式は P_t/P_{t-1} を決定するもので P_{t-1}, P_t は決まらない。
  - （まぁ，値に意味はないのだけど……）
- 連鎖方式は漸化式
  - → 初期値の情報 $P_0 = 1$ が決まれば順々に決まっていく。

ラスパイレス指数

$$P_0 = 1$$

$$P_t = P_{t-1} \times \frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_{t-1}}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_{t-1}}$$

$$t = 1, 2, \ldots$$

パーシェ指数

$$P_0 = 1$$

$$P_t = P_{t-1} \times \frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_t}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_t}$$

$$t = 1, 2, \ldots$$

特に重要なのはパーシェ式の連鎖指数で，（概念上は）このように計算される指数の例に GDP デフレーターがある。（実務上はかなり違う）

フィッシャー指数

- 統計データ作成の実務では多段階の指数計算が行われる。
  1. 最細目単位で指数計算，
  2. 最細目の指数を平均して細目単位の指数計算，
  3. ……

  米と麦（品種ごとに価格変化が異なる）の価格指数をそれぞれ計算したあとに，これらを合わせて穀類の価格指数を計算する，など。
- 各段階ではフィッシャー指数も使われる。ラスパイレス式とパーシェ式の相乗平均を用いて次のように定義される。

$$\frac{P_t}{P_{t-1}} = \left(\frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_{t-1}}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_{t-1}} \times \frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_t}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_t} \right)^{1/2}$$

固定基準年方式

- 家計の生計費に関係する物価指数「消費者物価指数」(CPI) では，連鎖方式は「参考指数」として副次的な取り扱い。
- CPI は「固定基準年方式」のラスパイレス式指数を採用している。次のように定義する。

$$\frac{P_t}{P_{t-1}} = \frac{\boldsymbol{p}_t \cdot \boldsymbol{x}_0}{\boldsymbol{p}_{t-1} \cdot \boldsymbol{x}_0}$$

- $t \neq 1$ 期の指数 P_t は $t-1$ 期のデータに依存しない。→ 確認してみよう。

指数の数値計算

連鎖指数を決定するために必要なデータ

前期の指数 P_{t-1} と，価格ベクトルと数量ベクトル

$$\boldsymbol{p}_{t-1} = \begin{bmatrix} p_{t-1,1} \\ \vdots \\ p_{t-1,n} \\ \vdots \\ p_{t-1,N} \end{bmatrix} \quad \boldsymbol{p}_{t} = \begin{bmatrix} p_{t,1} \\ \vdots \\ p_{t,n} \\ \vdots \\ p_{t,N} \end{bmatrix} \quad \boldsymbol{x}_{t-1} = \begin{bmatrix} x_{t-1,1} \\ \vdots \\ x_{t-1,n} \\ \vdots \\ x_{t-1,N} \end{bmatrix} \quad \boldsymbol{x}_{t} = \begin{bmatrix} x_{t,1} \\ \vdots \\ x_{t,n} \\ \vdots \\ x_{t,N} \end{bmatrix}$$

ベクトルは `np.array()` で定義できる。

例

2000 年の指数を 1 として，2001 年のラスパイレス価格指数とパーシェ価格指数を計算しなさい。

	A 財		B 財	
	価格	数量	価格	数量
2000 年	40	3	80	6
2001 年	80	5	30	7

計算例

```
price00 = np.array([40, 80])
price01 = np.array([80, 30])
quantity00 = np.array([3, 6])
quantity01 = np.array([5, 7])

np.sum(price01* quantity00) / np.sum(price00 * quantity00)
# Laspeyres
```

```
0.7
```

```
np.sum(price01 * quantity01) / np.sum(price00 * quantity01)
# Paasche
```

```
0.8026315789473685
```

めんどくさい？

- よいプログラマーになるには怠惰でなければならないらしい。
- このコードは 2 年分だからいいけど，5 年，10 年，100 年データがあったときに，各年度の価格と数量を表す変数を作るなんてことは，絶対にやりたくない。
- 表を書き直して頭の中を整理しよう。

例’

次のように変形できる。

価格	A 財	B 財
2000 年	40	80
2001 年	80	30

数量	A 財	B 財
2000 年	3	6
2001 年	5	7

データは行列形式で表現できる。→ 変数は 2 つで済む

$$\text{価格} = \begin{bmatrix} 40 & 80 \\ 80 & 30 \end{bmatrix}, \quad \text{数量} = \begin{bmatrix} 3 & 6 \\ 5 & 7 \end{bmatrix}$$

行番号 0, 1 が 2000 年，2001 年に，列番号 0, 1 が A 財，B 財に対応。
こういう対応関係はプログラムの外で決めておくのでもいいし，行・列のラベリングができる環境（Python なら pandas）を使ってもいい。

行列も np.array()

2 重の角かっこに注意

行列は「リストのリスト」に np.array()

```
price = np.array([[40, 80],
                  [80, 30]])
quantity = np.array([[3, 6],
                     [5, 7]])
```

価格	A 財	B 財
2000 年	40	80
2001 年	80	30

数量	A 財	B 財
2000 年	3	6
2001 年	5	7

行列 [行，列]

```
price[0, 1]
```

```
80
```

```
price[:, 0]   # : は「全部」
```

```
array([40, 80])
```

```
quantity[1, :]
```

```
array([5, 7])
```

40	80
80	30

40	80
80	30

3	6
5	7

結果を格納する array

ベクトル・行列を作るときに便利な関数。
カッコの中の数字はサイズ

```
np.ones(2)
```

```
array([1., 1.])
```


```
np.zeros(2)
```

```
array([0., 0.])
```


```
np.empty(2)
```

```
array([0., 0.])
```

ラスパイレス指数

```
P = np.empty(2)
P[0] = 1.0
P[1] = P[0] * (np.sum(price[1, :] * quantity[0, :])
                / np.sum(price[0, :] * quantity[0, :]))
P
```

```
array([1.  , 0.7])
```

0, 1 が 少し煩わしい。

ラスパイレス指数 (改良版)

```
P = np.empty(2)
P[0] = 1.0
t = 1
P[t] = P[t-1] * (np.sum(price[t, :] * quantity[t-1, :])
                 / np.sum(price[t-1, :] * quantity[t-1, :]))
P
```

```
array([1.  , 0.7])
```

これで数式とほぼ同じになった！　L5-6 は t の値によらず同じ。

問題：パーシェ指数

やり残した仕事：

2000 年の指数を 1 として，2001 年の**パーシェ価格指数**を計算しなさい。

	A 財		B 財	
	価格	数量	価格	数量
2000 年	40	3	80	6
2001 年	80	5	30	7

問題：長い期間のデータ

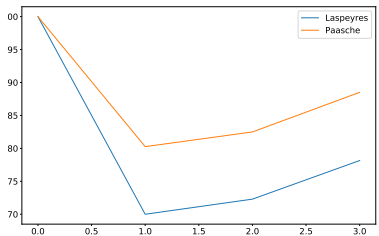
2000 年の指数を 1 として**ラスパイレス価格指数の系列** (PL) と**パーシェ価格指数の系列**（PP）を計算しなさい。

価格	A 財	B 財
2000 年	40	80
2001 年	80	30
2002 年	70	40
2003 年	60	55

数量	A 財	B 財
2000 年	3	6
2001 年	5	7
2002 年	6	8
2003 年	8	10

結果のプロット

```
import matplotlib.pyplot as plt
plt.plot(PL, label = 'Laspeyres')
plt.plot(PP, label = 'Paasche')
plt.legend()
plt.show()
```



経済理論にもとづかない数字を使ったのでおかしくなってしまったけど，
本当ならラスパイレス指数の方がパーシェ指数より大きくなる傾向がある。→理由を考えてみよう。（ミクロの教員に聞いてみよう）