課題２　包絡線定理

花嶋 陽

2014/6/7

1　はじめに

包絡線定理は、経済学において効用最大化や費用最小化等の最大化、最小化問題を解く際に活用されます。以下では、単純な関数で包絡線定理について説明した後、包絡線を描画するプログラムのコードについて解説します。

2　包絡線定理

関数

$$f(x,t) = tx - t^2$$

が与えられているとします。t をパラメータと見て、x − y 平面上の直線

$$l_t : y = tx - t^2$$

を考えると、t の値を変化させるごとに１本の直線が引けます。t の値をいろいろ変えて直線 l_tをいくつも描いたものが図１、図２です。図を見ると、直線 l_tの通過領域はある曲線（C と呼ぶことにします）の下側全体となっていることが分かります。この曲線を表す関数を $F(x)$ とおくと、

$$F(x) = \max_t f(x,t)$$

となります。ここで f を t について平方完成すると

$$f(x,t) = -\left(t - \frac{x}{2}\right)^2 + \frac{x^2}{4}$$

となるので、f は t = x/2 で最大値 $\mathrm{x}^2/4$ となることがわかります。よって、

$$F(x) = \frac{x^2}{4}$$

となります。次に、図より曲線 C と各 l_tとが必ず接する形で交点を持つことが見てとれますが、これを確かめましょう。先の t についての最大化問題の解をt*(x)と置きます。各 $x = \bar{x}$

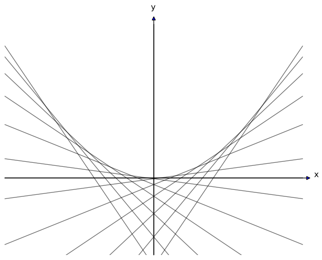

図 1: 接線の本数少なめ

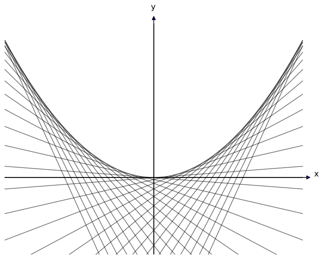

図 2: 接線の本数多め

において、F() = f(t*(),)なので、$x = \bar{x}$で曲線 C と直線 $l_{t*(\bar{x})}$は交わります。また、C の傾きは $F'(\bar{x}) = \bar{x}/2$ であり、直線 $l_{t*(\bar{x})}$は

$$y = \frac{\bar{x}}{2}x - \frac{x^2}{4}$$

となるので、その傾きも $\bar{x}/2$ であり、たしかに曲線 C と直線 $\mathrm{l}_{t*(\bar{x})}$は任意の x において接することが分かりました。以上のことから、$F'(x)$ を計算するのに、

$$\frac{\partial f}{\partial x}(t*(x), x)$$

を計算すればよいことがわかります。つまり、関数 f(x,t) について、

$$F(x) = \max_t f(x, t)$$

とし、F(x) と f(x, t) は x について微分可能だとします。ここで、各 x に対して最大値を与える t を t*(x) と表します。このとき、

$$F(x) = \frac{\partial f}{\partial x}(t*(x), x)$$

が成り立ちます。これを包絡線定理と言います。

3 Python プログラム

```
# -*- coding: utf-8 -*-

import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.axes_grid.axislines import SubplotZero


# 図の背景の諸体裁を設定

# 作図スペースを用意。
fig = plt.figure(1)
ax = SubplotZero(fig, 111)
fig.add_subplot(ax)

# 軸の設定
ax.axhline(linewidth=1.2, color="black")
ax.axvline(linewidth=1.2, color="black")

# 軸に矢印
for direction in ["xzero", "yzero"]:
ax.axis[direction].set_axisline_style("-|>")
ax.axis[direction].set_visible(True)

# 四方の軸を消す。
for direction in ["left", "right", "bottom", "top"]:
ax.axis[direction].set_visible(False)

# 軸に名前を付ける。位置は適宜設定。
plt.figtext(0.93, 0.37, 'x')
plt.figtext(0.505, 0.95, 'y')

# 軸の目盛を消す。表示する y 軸の範囲を設定 (グラフが見易くなるよう適宜設定)。

plt.xticks([])
plt.yticks([])
plt.ylim(-3.5,6.5)


# 図を描く条件設定
```

```
# 元になる関数を定義
def f(x, t):
    return t * x - t**2

# 変数
savever = 'png' # 'png' or 'pdf'
fignum = 0 # 0 or 1

if fignum == 0:
    p = 5    #x の範囲  -p<=x<=p（左右対称にするため）
    r = 2    #t の範囲  -r<=t<=r（同上）
    n = 12   #接線の本数

if fignum == 1:
    p = 5
    r = 3
    n = 30

# 包絡線を作る
x = np.linspace(-p,p,2)    # 直線なのでプロットする点は 2 点でいいかと。
t = np.linspace(-r,r,n)  # 傾きは n-1 等分で均等に。lispace で範囲内を n-1
等分した array を用意。
for i in t:
y = f(x, t=i)
ax.plot(x, y, 'k-', linewidth=1.0, alpha=0.6)
plt.savefig('envelope' + str(fignum) + '.' + savever)
plt.show()
```

工夫した所は、t の値を linspace による array で用意したことです。
課題としては、y の表示の範囲を変数の変更に応じて自動で調整できればと思います。

参考文献

[1] 尾山大輔・安田洋祐「経済学で出る包絡線定理」『経済セミナー』2011 年 10・11 月号.