Zadanie Egzaminacyjne RPiS

Sprawozdanie do zadania **L1Z4**

Filip Komorowski

11 kwietnia 2021

1 Wstęp

Rozkład t-studenta to rozkład prawdopodobieństwa stosowany przy konstruowaniu przedziałów ufności, testowaniu hipotez statystycznych oraz do oceny błędu pomiaru.

Rozkład t studenta stosujemy raczej tylko w sytuacji gdy odchylenie standardowe populacji jest nieznane, a liczba obserwacji jest mniejsza niż 30. W przypadku gdy rozmiar próby jest większy lub równy 30 wtedy zamiast brać rozkład t bierzemy rozkład normalny. Wynika to z faktu, że rozkład t studenta dla $n \geqslant 30$ jest bardzo podobny do rozkładu normalnego. Dla $n < 30$ rozkład studenta jest „szerszy", tzn. bardziej prawdopodobne są wartości mocno odbiegające od średniej niż w przypadku rozkładu normalnego.

W tej pracy spróbujemy przyjrzeć się funkcji gęstości tego rozkładu, a dokładniej podejmiemy się zadania policzenia całki z funkcji gęstości rozkładu t-Studenta o k stopniach swobody.

Funkcja gęstości:

$$f(x) = \frac{\Gamma(\frac{k+1}{2})}{\sqrt{k\pi}\,\Gamma(\frac{k}{2})}(1 + \frac{x^2}{k})^{-\frac{k+1}{2}}$$

Interesująca nas całka:

$$\int_{-\infty}^{t} f(x)\,dx$$

2 Obserwacje

2.1 Symetryczność

Zauważmy, że w naszej funkcji gęstości x występuje tylko w jednym miejscu i jest on podnoszony do kwadratu. Z tego wynika, że $f(x) = f(-x)$, czyli funkcja jest symetryczna.

A mając na uwadze dodatkowo fakt, że f jest funkcją gęstości to możemy zauważyć, że

$$\int_{-\infty}^{0} f(x)dx = \frac{1}{2}$$

Zatem zależnie od tego, czy t jest dodatnie czy ujemne będziemy musieli policzyć odpowiednio:

$$\int_{-\infty}^{t} f(x)dx = \frac{1}{2} + \int_{0}^{t} f(x)dx$$

$$\int_{-\infty}^{t} f(x)dx = \frac{1}{2} - \int_{0}^{t} f(x)dx$$

2.2 Całka trudna

Nawet po rozwiązaniu problemu nieoznaczoności całki, jest to w dalszym ciągu całka trudna do policzenia analitycznie. Będziemy zatem chcieli podejść do problemu numerycznie i skorzystamy z metod całkowania trapezów oraz Romberga, a do tego będziemy musieli wyliczać wartości f w wielu punktach.

Zanim przejdę do opisu tych metod, możemy się zastanowić czy możemy coś zrobić z nieprzyjemnie wyglądającym ułamkiem $\frac{\Gamma(\frac{k+1}{2})}{\sqrt{k\pi}\,\Gamma(\frac{k}{2})}$ co ułatwiłoby nam wyliczanie f.

Jak wiemy z ćwiczeń:

$$\Gamma(\tfrac{1}{2}) = \sqrt{\pi}$$

$$\Gamma(n) = (n-1)!$$

Dodatkowo przydatna okaże się też taka własność:

$$\Gamma(n + \tfrac{1}{2}) = \tfrac{(2n-1)!!}{2^n}\sqrt{\pi}$$

Wtedy dla ustalonego k możemy prosto wyliczyć sobie cały ułamek:
Gdy k jest parzyste:

$$\frac{\Gamma(\frac{k+1}{2})}{\sqrt{k\pi}\,\Gamma(\frac{k}{2})} = \frac{\sqrt{\pi}\,\frac{(k-2)!!}{2^{\frac{k-1}{2}}}}{\sqrt{k\pi}(\frac{k}{2}-1)!} = \frac{\frac{(k-2)!!}{2^{\frac{k-1}{2}}}}{\sqrt{k}(\frac{k}{2}-1)!}$$

Gdy k jest nieparzyste:

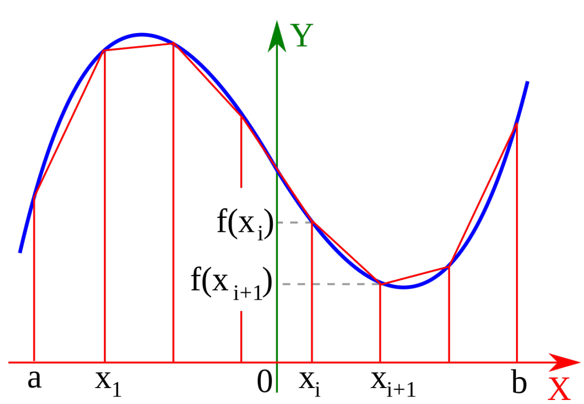
$$\frac{\Gamma(\frac{k+1}{2})}{\sqrt{k\pi}\,\Gamma(\frac{k}{2})} = \frac{(\frac{k+1}{2}-1)!}{\sqrt{k\pi}\,\frac{(k-2)!!}{2^{\frac{k-1}{2}}}\sqrt{\pi}} = \frac{(\frac{k+1}{2}-1)!}{\pi\sqrt{k}\,\frac{(k-2)!!}{2^{\frac{k-1}{2}}}}$$

3 Metody Całkowania

3.1 Metoda Złożonych Trapezów

W tej metodzie szacujemy całkę, czyli pole pod wykresem funkcji, poprzez podzielenie go na trapezy i licząc sumę ich pól.

Jeżeli przedział całkowania $[a, b]$ podzielony zostanie punktami $x_1, x_2, \ldots, x_{n-1}$ na n równych części o długościach $\frac{b-a}{n}$ i w figurę ograniczoną prostymi $x = a, x = b$, osią x oraz wykresem funkcji $y = f(x)$ wpiszemy trapezy jak pokazano na rysunku poniżej:



To pola kolejnych trapezów wynoszą $\frac{b-a}{n}\frac{f(x_0)+f(x_1)}{2}, \frac{b-a}{n}\frac{f(x_1)+f(x_2)}{2}, \ldots, \frac{b-a}{n}\frac{f(x_{n-1})+f(x_n)}{2}$ a ich suma szacuje całkę $\int_a^b f(x)dx$

Mój kod implementujący tę metodę:

```python
def trapezy(n):
    result = 0
    h = abs(t) / n
    for i in range(n):
        x = i * h
        result += h * (f(x) + f(x + h))/2
    if t < 0:
        return 0.5 - result
    return result + 0.5
```

3.2 Metoda Romberga

W metodzie Romberga dzielimy przedział całkowania $[a, b]$ na 2^i równych części i wyliczamy $f(x_i)$ dla tych punktów. Dodatkowo niech $h_i = \frac{b-a}{2^i}$ oznacza długość kroku.

Teraz metodę Romberga możemy zapisać rekurencyjnie w następujący sposób:

$$\begin{cases} \mathrm{R}_{0,i} : R_{2^i} = h_i * \sum_{k=0}^{2^i-1}\left(\frac{f(x_k)+f(x_{k+1})}{2}\right) \\ \mathrm{R}_{m,i} : \frac{4^m R_{m-1,i+1} - R_{m-1,i}}{4^m - 1} \end{cases}$$

Gdzie $R_{0,i}$ jest wzorem złożonych trapezów. Po obliczeniu pierwszej kolumny tablicy Romberga, kolejne wyliczane są rekurencyjnie, otrzymując coraz lepsze przybliżenie funkcji.

Moja implementacja tej metody:

```python
def romberg(n):
    h = [0]*(n+1)
    r = [[0]*(n+1) for i in range(n+1)]
    for i in range(1, n + 1):
        h[i] = abs(t) / pow(2, i-1)

    r[1][1] = h[1] / 2 * (f(0) + f(t))

    for i in range(2, n + 1):
        wsp = 0
        for j in range(1, pow(2, i-2) + 1):
            wsp += f((2*j - 1) * h[i])
        r[i][1] = 0.5 * (r[i-1][1] + h[i-1] * wsp)

    for i in range(2, n+1):
        for j in range(2, i+1):
            r[i][j] = r[i][j-1] + (r[i][j-1]-r[i-1][j-1]) / (pow(4, j-1)-1)

    if t < 0:
        return 0.5 - r[n][n]
    return r[n][n] + 0.5
```

4 Wyniki przeprowadzonych eksperymentów

Dla każdego przykładu moim punktem odniesienia jest: https://www.statskingdom.com/t-student.html z wynikiem liczonym do 9 cyfr znaczących.

4.1 $k = 2, t = 0.5$

Oczekiwana wartość całki: 0.666666667

Wynik mojego programu:

```
Dla k = 2 i t = 0.5:
Wynik dla trapezów: 0.6666505902714797 przy 16 punktach.
Wynik dla romberga: 0.6666666666315029 przy 16 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.6666626478101896 przy 32 punktach.
Wynik dla romberga: 0.6666666666669616 przy 32 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.6666656619676894 przy 64 punktach.
Wynik dla romberga: 0.6666666666666664 przy 64 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.6666664154928686 przy 128 punktach.
Wynik dla romberga: 0.6666666666666666 przy 128 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.6666666038732763 przy 256 punktach.
Wynik dla romberga: 0.6666666666666666 przy 256 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.6666666509683228 przy 512 punktach.
Wynik dla romberga: 0.6666666666666665 przy 512 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.666666662742081 przy 1024 punktach.
Wynik dla romberga: 0.6666666666666667 przy 1024 punktach
------------------------------------------------------
Wynik dla trapezów: 0.6666666656855202 przy 2048 punktach
Wynik dla romberga: 0.6666666666666666 przy 2048 punktach
```

4.2 $k = 2, t = 0.9$

Oczekiwana wartość całki: 0.7684474938

Wynik mojego programu:

```
Dla k = 2 i t = 0.9:
Wynik dla trapezów: 0.7683937037937199 przy 16 punktach.
Wynik dla romberga: 0.7684474147399175 przy 16 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684340472440333 przy 32 punktach.
Wynik dla romberga: 0.7684474940970213 przy 32 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684441322358533 przy 64 punktach.
Wynik dla romberga: 0.7684474938221056 przy 64 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684466534293577 przy 128 punktach.
Wynik dla romberga: 0.7684474938223522 przy 128 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684472837243304 przy 256 punktach.
Wynik dla romberga: 0.7684474938223522 przy 256 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684474412978607 przy 512 punktach.
Wynik dla romberga: 0.7684474938223522 przy 512 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684474806912303 przy 1024 punktach.
Wynik dla romberga: 0.768447493822352 przy 1024 punktach.
------------------------------------------------------
Wynik dla trapezów: 0.7684474905395712 przy 2048 punktach.
Wynik dla romberga: 0.7684474938223521 przy 2048 punktach.
------------------------------------------------------
```

4.3 $k = 3, t = 0.25$

Oczekiwana wartość całki: 0.5906353883

Wynik mojego programu:

```
Dla k = 3 i t = 0.25:
Wynik dla trapezów: 0.5906330456234495 przy 16 punktach.
Wynik dla romberga: 0.5906353887860961 przy 16 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906348030049874 przy 32 punktach.
Wynik dla romberga: 0.5906353887855851 przy 32 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906352423410567 przy 64 punktach.
Wynik dla romberga: 0.5906353887855852 przy 64 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906353521744919 przy 128 punktach.
Wynik dla romberga: 0.5906353887855853 przy 128 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906353796328144 przy 256 punktach.
Wynik dla romberga: 0.5906353887855853 przy 256 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906353864973926 przy 512 punktach.
Wynik dla romberga: 0.5906353887855852 przy 512 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906353882135371 przy 1024 punktach.
Wynik dla romberga: 0.5906353887855853 przy 1024 punktach.
-----------------------------------------------------
Wynik dla trapezów: 0.5906353886425733 przy 2048 punktach.
Wynik dla romberga: 0.5906353887855853 przy 2048 punktach.
```

5 Podsumowanie

Widzimy, że metoda Romberga okazuje się znacznie efektywniejsza od metody złożonych trapezów. Oczekiwaną dokładność 8-miu miejsc po przecinku uzyskujemy w przypadku trapezów dopiero przy średnio tysiącu punktach. Gdy w przypadku metody Romberga często wystarczy ich tylko szesnaście.