Robin Karlose
40089313

Problem 1

1.Function Definition

The Beta function is a special function in Mathematics, which is defined by the following formula
$B(x, y) = \int_0^1 t^{x-1}(1-t)^{y-1}dt$
The domain of the function is for – Re x > 0, Re y > 0. (Positive Real numbers x and y)
The co-domain of the function is also the same Re x > 0, Re y > 0. (Positive Real numbers x and y)
It is also known as the Euler integral of the first kind. The Beta function is important in Mathematics , calculus and analysis due to its relationship with the gamma function and the factorial function. Interestingly a lot of complex integrals can be derived and reduced to simpler expressions involving the Beta function

2. Characteristics of the Beta Function

- $B(m, n) = B(n, m)$
- $B(m, n) = 2\int_0^{\frac{\pi}{2}} \sin^{2m-1}\Theta \cos^{2n-1}\Theta d\Theta$
- $B(m, n) = \int_0^{\infty} \frac{x^{m-1}}{(1+x)^{m+n}}dx$
- $B(m, n) = \int_0^{1} \frac{x^{m-1}+x^{n-1}}{(1+x)^{m+n}}dx$

It is also related to the Gamma Function with the following relationship
$B(p, q) = \frac{\Gamma p \Gamma q}{\Gamma(p+q)}$
Also it possesses the Recurrence Relationship property
$B(x+1, y) = B(x, y)\frac{x}{x+y}$
Relationship with factorial method
$B(x, y) = \frac{(x-1)!(y-1)!}{(x+y-1)!}$

Functional Requirements

Identifier - FR1

FR1 — The Beta Function can provide an output if it has an input from a domain comprising of positive Real Numbers only.

Identifier - FR2

FR2 — The Beta Function , can only provide an output if its parameters , X and Y are both positive numbers i.e. Real Number X > 0 and Real Number Y > 0.

Identifier - FR3

FR3 — To compute the value of the Beta Function , a subordinate function needs to be used to calculate the value of A raised to the power B. In other words we need to define a power function to calculate A^B .

Identifier - FR4

FR4 —To computer the value of the Beta Function for any Real number, we need to be able to compute the Definite Integral as defined in the mathematical realm of Calculus.

Functional Assumption

FA1 — To compute the value of the Beta Function , we can estimate the value of the Definite Integral using Numerical Methods.

NON FUNCTIONAL REQUIREMENTS

Identifier - NFR1

NFR1 — To accurately compute the value of Beta Function for larger input values of X and Y , we need the ability to store very large decimal values.

Identifier - NFR2

NFR2 — The method used to calculate the Beta Function , should be scalable for different input values and different Hardware Requirements.

Identifier - NFR3

FR3 — The method used to calculate the Beta Function , should be optimized for performance so that it efficiently calculates the integral for large input values of X and Y.

Robin Karlose
40089313

Problem 3

The Beta Function

It is easier to define a generalized function called the incomplete Beta function The incomplete beta function is defined by

$$I_x(a,b) = \frac{x^a(1-x)^b}{aB(a,b)} \frac{1}{1+\frac{d_1}{1+\frac{d_2}{1+\frac{d_3}{1+\dots}}}}$$

There are several ways to solve the incomplete Beta function 2 ways would be as follows

First convert the Incomplete Beta function into a continued fraction

Algorithm 1

Evaluation of continued fraction using modified Lentz method

Algorithm 2

Evaluation of continued fraction using Gaussian quadrature rule

I choose **Algorithm 1** for the technical reasons discussed in the next section

Technical Reasons and Explanation of the Algorithm

Advantages

1.No actual integration required

2.Slightly Faster performance

3.It is standalone and does not require computation of the Gamma function

4.It will work for all positive real numbers (not only integers)

Disadvantages

1.It is an approximation and hence not too accurate

2.It can cause overflow or underflow of the floating point variable

3.It is a mathematically derived estimation

Explanation of Algorithm in short

We can express the Beta function in short like this

$$I_x(a,b) = \frac{x^a(1-x)^b}{aB(a,b)} \frac{1}{1+\frac{d_1}{1+\frac{d_2}{1+\frac{d_3}{1+\dots}}}}$$

Then we use Lentz algorithm to solve the continued fraction

$$F = 1 + \cfrac{a_1}{1 + \cfrac{a_2}{1 + \cfrac{a_3}{1 + \cfrac{a_4}{1 + \ldots}}}}$$

So , now as we have

$\mathrm{D}_0 = 0$
$C_0 = 1$
$F_0 = 1$

$\mathrm{D}_i = \frac{1}{1 + a_j D_{i-1}}$
$C_i = 1 + \frac{a_j}{C_{i-1}}$
$F_i = F_{i-1} C_i D_i$

We should iterate until C*D becomes really small We also have to make sure that C and D do not actually become 0 (or else it will become undefined). We could implement this by assigning C or D to a very small value if either C or D become too close to zero

Pseudocode

Algorithm 1: Incomplete Beta function using modified Lentz method

```
Function betacf(constant Doub a, constant Doub b, constant Doub x)
{
Variable Declarations
Integer m,m2;
Double  aa,c,d,del,h,qab,qam,qap;

qab = a+b
qap = a +1.0
qam = a-1.0

c=1.0
d=1.0-qab*x/qap

if (mod(d) < Floating Point minimum)
    d=Floating point minimum

d=1.0/d
h=d
```

```
for m = 1 to 9999
{
m2=2*m;
aa=m*(b-m)*x/((qam+m2)*(a+m2))
d=1.0+aa*d

if(mod(d) < Floating Point minimum)
    d= Floating Point minimum

c=1.0+aa/c

if (mod(c) < Floating Point minimum)
    c= Floating Point minimum
d=1.0/d
h *= d*c

aa = -(a+m)*(qab+m)*x/((a+m2)*(qap+m2))

d=1.0+aa*d

if (mod(d) < Floating Point Minimum)
    d=Floating Point Minimum;

c=1.0+aa/c

if (mod(c) < Floating Point Minimum)
    c=Floating Point Minimum

d=1.0/d
del=d*c
h *= del

if (mod(del-1.0) <= EPS)
    break

m=m + 1
}

return h
}
```

References

William H Press,Saul Teukolksky, WIlliam T .Vetterling,Brian P Flannery, The Art of Scientific Computing , (3rd Edition , Cambridge Press), Chapter 6 , Pg 270 - 273

Abramowitz, M., and Stegun, I.A. 1964, Handbook of Mathematical Functions (Washington: National Bureau of Standards); reprinted 1968 (New York: Dover); online at http://www.nr.

com/aands, Chapters 6 and 26.[1]

Pearson, E., and Johnson, N. 1968, Tables of the Incomplete Beta Function (Cambridge, UK: Cambridge University Press).