


| 카카오톡 봇 강좌 | >

[응용] 함수를 적분하는 방법에 대해서.araboza



천비아

챗봇 입문자

1:1 채팅

2019.09.15. 13:22 조회 165

댓글 13 URL 복사

강좌할 주제

=> 수치해석과 정적분

강좌 내용

=> 아래에 있습니다

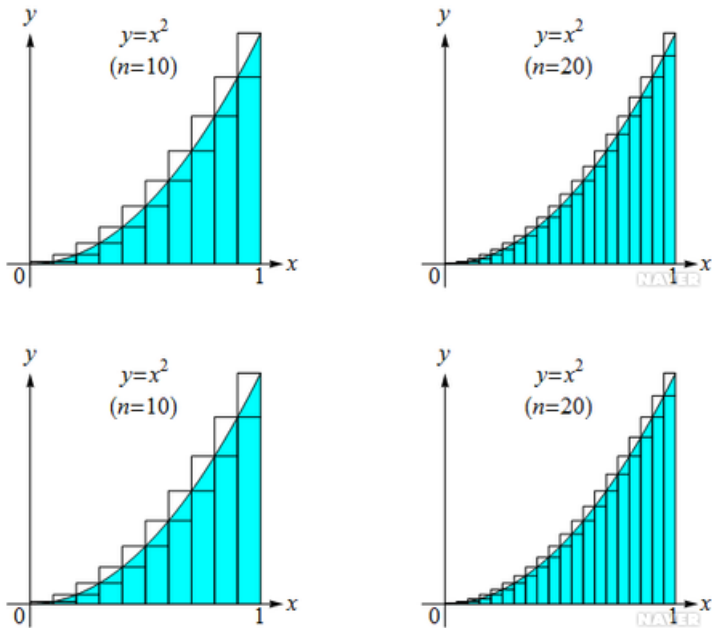
(말머리 등록 하셨나요? Yes / No)

0. 들어가기 전

적분이 뭔가요? -> 모르면 살포시 뒤로가기...
앵? 다항함수 적분 그거 쉬운거 아니냐? -> 물론 미적분학의 기본정리(FTC)를 사용하게 되면 간단한 다항함수의 적분에 대해 결과값을 얻을 수 있겠지만, 다른 함수들을 처리하지 못하게 됩니다. 예를 들면, 내가 "e^x"라는 아주아주 기본적인 초월함수를 구현해냈는데 미분도 적분도 못하면 말이 되겠습니까?

1. Rectangle Method

학창시절에 "구분구적법"이라는 이름으로 공부해본 바로 그 방법입니다.



바로 사각형을 이용하는 방법입니다.

그런데 이건 리미트도 들어가야하고 상합과 하합을 구해야 하는 불편함이 있더군요?

수치적분을 하는데 그런건 필요 없습니다!

우리가 원하는 것은 단순한 "근삿값"일 뿐이지, 정확한 값을 구하고자 하는 것이 아닙니다.

물론 컴퓨터를 무한시간동안 rectangle method를 실행시켜주면 되긴 하겠지만... 무한 시간동안 작동하는 프로그램이 쓸모가 있나요?

어쨌든, 이를 간단하게 구현해보면 다음과 같습니다.

```
Rect: function(F, a, b, n)
{
    var sum = 0;
    var h = (b-a)/n;

    for(i = 0; i < n; i++)
    {
        sum += F(a+h/2+i*h)*h;
    }
    return sum;
}
```

구간의 크기를 n 개로 잘라주고, 그 구간의 길이를 h 로 설정한다음, 직사각형들을 만들어주고 더했습니다. 이 방법은 매우 간단하고 속도도 빠르지만, 정확도가 꽤나 떨어진다는 함정이 있습니다.

그렇다면, 무슨 방법을 써야할까요?

2. Trapezoidal Method

사각형이 안되면 사다리꼴은 어떨까요? 사각형은 위아래로 찢리는게 좀 많으니까, 각각의 구간간에 대해 선형근사하여 사다리꼴을 만들어주는 겁니다. 이렇게요.



사진은 전부 네이버에서 그때그때 찾아오고 있습니다(?)

어쨌든 이 방법을 "간단히" 구현하면, 다음과 같습니다.

```
Trap: function(F, a, b, n)
{
    var sum = 0;
    var h = (b-a)/n;

    for(i = 0; i < n; i++)
    {
        sum += (F(a+i*h)+F(a+(i+1)*h))/2*h;
    }
    return sum;
}
```

그러나 이는 아주 간단히 구현한 방식이기 때문에, 가끔 위의 Rectangle Method보다 오차가 커지게 됩니다.(세상에!)

수정하는 방법을 이용할 수도 있긴 한데... 여기서 수치해석 강좌까지 쓰고싶지는 않아서(?) 더 좋은 방법을 소하려고 합니다.

3. Simpson's Method

그 심슨 아니고 사람 심슨입니다.

옐 또 사진이 뭐이리 커..

심슨의 방법은 매우매우 쉬우면서 오차도 매우매우 적습니다! 구현하기도 쉬워요.

다른 방식이(Rect, Trap) 일차근사(선형근사)하여 함수의 적분을 구하는 방식이었다면, 이 방식은 이차함수로 다항근사하는 방식입니다. 즉, 일차함수보다 훨씬 정확가 높습니다.

그러면서 구현도 쉬우니, 코드 한 번 보시죠.

```
Simp: function(F, a, b, n)
{
    var sum = 0;
    var h = (b-a)/n;

    for(i = 0; i < n; i++)
    {
        sum += (F(a+i*h)+4*F(a+i*h+h/2)+F(a+(i+1)*h))*h/6;
    }
}
return sum;
}
```

이런 괴상한 식이 과연 적분을 제대로 할 수 있을까 싶겠지만, 실제로 수학적으로도 증명이 된 내용이므로 믿으셔도 좋습니다.

그런데 일차근사보다 이차근사가 정확도가 높으면, 삼차근사는 이차근사보다 정확할까요?

아쉽게도 항상 그런 것은 아닙니다. 이차근사는 일차근사보다 우월하지만, 삼차근사가 이차근사보다 항상 우월하지는 않습니다. 즉, 여기서부터는 정확도를 높이려면 "차수"를 높이는 것보다 "n"을 늘려서 구간을 더 잘게 쪼개는 것이 좋다는 것이지요.

무작정 구간을 늘리는 것이 정말 최선일까요?

4. Adaptive Quadrature Method

이 방법은 앞서 소개한 3개의 방법만큼 간단치는 않습니다. 우선 코드부터 보고 분석하도록 하겠습니다.

```
Adapt: function(F, a, b, tolerance)
{
    var x = Simp(F, a, b, 4);

    if(fabs(x - Simp(F, a, b, 2)) > tolerance)
    {
        return Adapt(F, a, (a+b)/2, tolerance) + Adapt(F, (a+b)/2, b, tolerance);
    }
}
return x;
}
```

tolerance라는 "오차 한계치"를 설정하여, 주어진 구간을 4개로 나누어 Simpson Method를 사용한 것과 2로 나누어 Simpson method를 사용한 것의

차이를 분석하고, 그 차이가 한계를 넘어가면 구간을 반으로 잘라 다시 계산하는 방법입니다.

실제로 수치적분은 함수의 "곡률"에 따라 그 오차가 생기게 되는데, Adaptive Method를 사용하면 이 곡률을 분석하여 오차를 줄여가므로 훨씬 오차가 적습니다.

왜 이렇게 하는지는 지는 곡률에 대한 이해가 필요한데, 곡률 없이 간단히 설명하면

"어떤 구간은 잘게 쪼개는게 오차가 적고, 어떤 구간은 잘게 쪼갤수록 오차가 커진다. 그러니까 각 부분에 대해 잘게 쪼개는게 나은지, 큼지막하게 자르는게 나은지 분석하여 수치적분하자."

라는 것입니다. 이러면 귀찮게 무한한 구간을 나누는 것보다 훨씬 빠르면서 정확도가 높아질 것 같군요!

5. 마치며

수학은 프로그래밍에서 뭘래야 뭘 수 없는 학문입니다. 물론 파싱을 해서 적분값을 가져오는 것도 좋은 방법이지만, 때로는 자신이 직접 적분을 구현해보는 것은 어떨까요? :)



천비아님의 게시글 더보기 >

❤️ 좋아요 1 💬 댓글 13

🔗 공유 | 신고

댓글 등록순 최신순 ↻

댓글알림 ☐



천비아 작성자

아 카페 글쓰기 진짜 구리다... 코드 다 빠져나왔네

2019.09.15. 13:26 답글 쓰기



바지

!!! 고인물 냄새

2019.09.15. 13:26 답글 쓰기



천비아 작성자

고인물이라뇨? 뉴비입니다만...

2019.09.15. 13:29 답글 쓰기



사로로

으악 4번째 방법 빼고는 다 알다니....

구분구적법은 이번 교육과정에서 사라졌다는 전설이...

2019.09.15. 13:29 답글 쓰기



천비아 작성자

맞아요. 되게 아쉽더라구요 ㅜㅜ 코딩을 넣는데 어떻게 구분구적법을 뭘 생각을 하지...

+ 혁... 대단하시네요. 3-8 Rule이나 Boole's Rule도 소개할 걸 그랬나요?

++ 것보다 Adaptive Method는 구현하기도 쉬우면서 1~3번보다 좋으니 애용해주세요 ㅎ

2019.09.15. 13:32 답글 쓰기



큐브



2019.09.15. 13:45 답글 쓰기



성빈

아라보자 ㄷ

2019.09.15. 14:24 답글 쓰기



scap

첫번째 방법에서

```
sum += F(a+h/2+i*h)*h;
```

이렇게 한 이유는 뭔가요

```
sum += F(a+i*h)*h;
```

이렇게 한다고 배우지 않나요?

h/2를 더한 이유는 뭔가요?

2019.09.15. 17:55 답글 쓰기



천비아 작성자

상합을 구해도 되고, 하합을 구해도 되지만 중간값을 구해도 됩니다. 중간값이 더 나은 상황이 꽤 되기 때문에 해당 방법을 사용했습니다. (상합이나 하합은 대부분의 상황에서 오차를 100% 가지고 갑니다.)

2019.09.15. 20:01 답글 쓰기



scap

천비아 아 중간값 구하는 거군요

2019.09.15. 20:02 답글 쓰기



SP

영 뭔소리지 역시 난 아직 초보군 부족해..

2019.09.15. 17:53 답글 쓰기



에샤마리

?!?!?!?!?!?!?!?! 적분?!?!?! 적부운?!?!?

2019.11.29. 03:55 답글 쓰기



hoy

천비아님 지금 있으신가요

2020.05.30. 10:44 답글 쓰기

Hibot

댓글을 남겨보세요



등록

글쓰기

답글

목록

▲ TOP

'카카오톡 봇 강좌' 게시판 글

전체 [응용] 말머리 글

이 게시판 새글 구독하기 ☐

브롤스타즈 api를 이용해 현재 및 다음맵을 가져와보자 🤖 [15]

사료로

2019.09.19.

1 2 3 4