

Riemann Sums

D. P. Morstad, University of North Dakota

Objectives of Assignment

1. To approximate the area between a graph and the x -axis using Riemann sums.
2. To write a subroutine which, given an arbitrary function, interval, and number of desired subintervals, will calculate the Riemann sums based on regular partitions.

I. Introduction

In mathematics it seems common to avoid doing hard problems by doing simpler ones instead. If a problem seems hard, don't solve it. Instead, turn it into a simple problem and then solve the simple problem.

To approximate the area under a curve, the problem is turned into a simple problem. Since it is easy to find the area of a rectangle, the region under the curve is first approximated by numerous adjacent rectangles. Then, to approximate the area under the curve, it is only necessary to add up the areas of the rectangles. The narrower the width of the rectangles, the better the approximation.

To find the exact value of the area under a curve, the Fundamental Theorem of Calculus will provide a technique which simplifies the problem. Determining the exact area under a curve is not the topic of this computer lab.

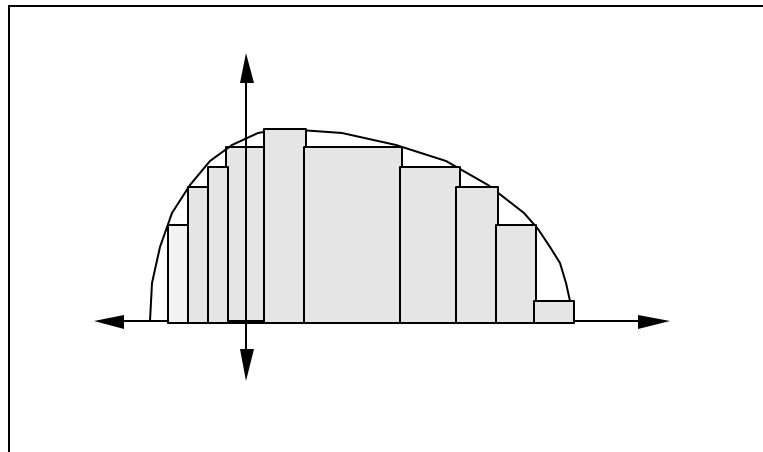


Fig 1. Area under the curve is approximated by numerous adjacent rectangles.

II. Finding the Areas of the Rectangles and their Sum

The area of a rectangle is simply its width times its length.

It is simplest to find the width of each rectangle if they all have the same width. For example, if you want the area under a curve from $x = 3$ to $x = 6$, and you want to use ten rectangles, just find the distance from 3 to 6 and then divide by ten. That is,

$$\text{width} = \frac{6-3}{10} \text{ or, in general, } \Delta x = \frac{b-a}{n}.$$

Subdividing an interval like this is called *partitioning* the interval. When each subinterval of the partition has the same width, it is called a *regular partition*.

The height of a rectangle must be the height of the function at some point in the subinterval. You can use the left endpoint of each subinterval, the midpoint, the right endpoint, or an arbitrary point of each subinterval. Each of the 10 subintervals (or n subintervals), will have its own point which we will denote by x_i^* . So, for example, the height of the third rectangle is $f(x_3^*)$.

The area of the third rectangle is:

$$A_3 = f(x_3^*) \cdot \Delta x.$$

The sum of the area of all ten rectangles (approximately the area under the curve) is:

$$A = \sum_{i=1}^{10} f(x_i^*) \cdot \Delta x.$$

III. An X(PLORE) Subroutine to do the Dirty Work

Since approximating the area between a curve and the x -axis boils down to finding $f(x_i^*)$ and multiplying it times Δx over and over and over again, and then adding it all up, this is a task which is ideally suited to a computer. It is quite easy to write a subroutine for X(PLORE) to do all this.

We will write a subroutine "leftsum($f(x)$, a , b , n)" which will base the height of each rectangle on its left endpoint, work for any function $f(x)$, and use n subintervals (n rectangles) over the interval $[a, b]$ on the x -axis.

After starting X(PLORE), press $\boxed{F10}$ to get to the subroutine screen, and then type the following subroutine. Do not type in any of the explanations in italics.

```

Procedure leftsum( $f(x)$ ,  $a$ ,  $b$ ,  $n$ )

procedure leftsum(% $f$ , $a$ , $b$ , $n$ )  $\Leftarrow$  name of the subroutine – notice the order of  $a$ ,  $b$ , and  $n$ 

erase
color(white)
graphics
}  $\Leftarrow$  clearing graphics screen, setting initial color, and setting graphics

graph( $f(x)$ , $x$ )  $\Leftarrow$  graphs the function (notice that a window is not defined in here)
 $dx = (b - a) / n$   $\Leftarrow$  finding the width  $\Delta x$  and calling it  $dx$ 
 $xi = a$   $\Leftarrow$   $x$  – coordinate of left end of first rect
 $area = 0$   $\Leftarrow$  total sum of the areas (which is 0 to start with)

for  $i = 1$  to  $n$  step 1  $\Leftarrow$  tells X(PLORE) to do this  $n$  times
   $rect(xi, f(xi), xi + dx, 0)$   $\Leftarrow$  graphs present rect (see manual)
   $area = area + f(xi) * dx$   $\Leftarrow$  adds area of this rect to total area so far
   $color((getcolor + 1) \bmod 15)$   $\Leftarrow$  changes color of next rect
   $xi = xi + dx$   $\Leftarrow$  moves to left end of next rect
end  $\Leftarrow$  go back to "for" and do it again
}  $\Leftarrow$  this is repeated once for each rect

text  $\Leftarrow$  takes X(PLORE) out of graphics auto
write('approx area is ',  $area$ )  $\Leftarrow$  reports total sum of area of all rects
end  $\Leftarrow$  end of subroutine

```

Pay close attention to the reason for each step and what it does. You will have to write a similar subroutine for the hand-in assignment.

III. Examples

```

Example 1. Approximate the area between  $f(x) = 9 - x^2$  and the  $x$ -axis from
 $x = -3$  to  $x = 3$ , using 10 subintervals, 25 subintervals, 250 subintervals.

Solution:
Leftsum does not define a window, but it tells X(PLORE) to graph  $f(x)$ 
along with several rectangles. You must first define the window, so, type in:

window(-5,5,-10,10)

```

$$f(x) = 9 - x^2$$

$$\text{leftsum}(f(x), -3, 3, 10)$$

Make sure you hit  for each line.

For 25 and 250, just change the 10 in leftsum.

10 subintervals approximates the area as 35.64.

25 subintervals approximates the area as 35.9424.

250 subintervals approximates the area as 35.99942.

Example 2. Approximate the area between $f(x) = \sin(x)$ and the x -axis:

a) on $[0, \mathbf{p}]$ using 20 subintervals, 500 subintervals.

b) on $[-2\mathbf{p}, 2\mathbf{p}]$ using 8 subintervals, 500 subintervals.

Solution: First, reset your window so it will include the entire interval and so it will graph the sine function with decent height.

a) Leftsum($f(x)$, 0, \mathbf{p} , 20).

20 subintervals approximates area as 1.995886.

500 subintervals approximates the area as 1.999959.

b) 8 subintervals approximates the area as 7.27596E-12

(this means $7.27596 \times 10^{-12} \approx 0$).

500 subintervals approximates the area as $-3.561610\text{E-}13 \approx 0$.

IV. Practice Problems

1. Approximate the area between the x -axis and $f(x) = \sin^2 x$ on $[0, 2\mathbf{p}]$, using 10 subintervals, 250 subintervals.
2. Approximate the area between the x -axis and $f(x) = x$ on $[0, 2]$, using 10 subintervals, 250 subintervals.
3. Approximate the area between the x -axis and $f(x) = \sec x$ on $[0, \frac{\mathbf{p}}{2}]$, using 10 subintervals, 250 subintervals.
4. Approximate the area between the x -axis and $f(x) = \sqrt{25 - x^2}$ on $[-5, 5]$, using 10 subintervals, 250 subintervals.
Same function, on $[-1, 1]$ using 10 subintervals, 250 subintervals.
5. Approximate the area between the x -axis and $f(x) = 3 \cos 2x$ on $[0, \frac{\mathbf{p}}{2}]$, using 10 subintervals, 250 subintervals.

Same function, on $[0, \mathbf{p}]$ using 10 subintervals, 250 subintervals.

6. Approximate the area between the x -axis and $f(x) = \frac{1}{x-1}$ on $[0, 2]$, using 10 subintervals, 250 subintervals.
7. Which lines in the subroutine tell X(PLORE) where the x_i^* is?
8. Which lines in the subroutine tell X(PLORE) what the rectangles are going to be?
9. Which lines in the subroutine tell X(PLORE) how to find the area of each rectangle? Why aren't these the same lines as for #8?