Math 174
Calculus II

Dr. Constant J. Goutziers
Department of Mathematics, Computer Science and Statistics
goutzicj@oneonta.edu

Lesson 14
Approximate Integration

Initializations

```
> restart;
  with(student):
>
```

14.1 Numerical Techniques

Examples

Example 14.1.1.

Approximate $\int_0^1 e^{-x^2}\,dx$.

Solution
Maple has a variety of methods for numerical integration. The methods related to calculus concepts are located in the student package.

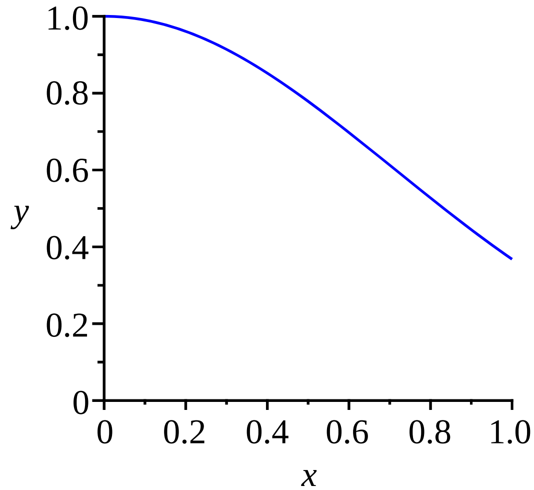
```
> f:=x->exp(-x^2);
  e1:=Int(f(x), x=0..1);
```

$$f := x \to e^{-x^2}$$

$$e1 := \int_0^1 e^{-x^2}\,dx \qquad (2.1.1.1)$$

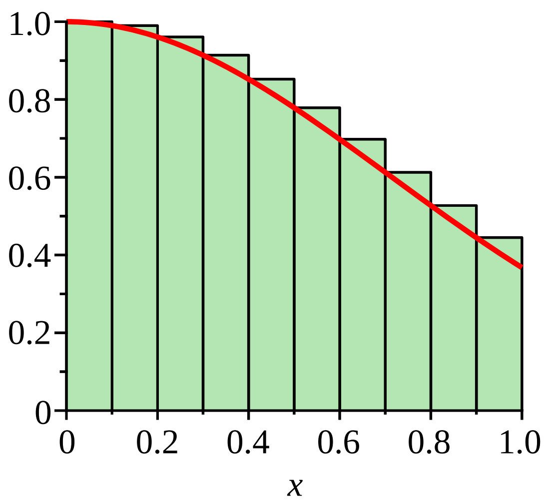
This integral represents the area under the curve:

```
> plot(f(x), x=0..1, y=0..1, color=blue);
```



We can approximate this area a by a sum of areas of the rectangles shown in the picture below.

```
> leftbox(f(x), x=0..1, 10);
```



This sum of areas is called a left sum and can be obtained by using the **leftsum** command.

```
> approximate_area_10:=leftsum(f(x), x=0..1, 10);
  approximate_area_10F:=evalf(approximate_area_10);
```

$$approximate\_area\_10 := \frac{1}{10} \sum_{i=0}^{9} e^{-\frac{1}{100} i^2}$$

$$approximate\_area\_10F := 0.7778168241 \qquad \textbf{(2.1.1.2)}$$

Of course we get a better approximation by using more rectangles.

```
> approximate_area_20:=leftsum(f(x), x=0..1, 20);
  approximate_area_20F:=evalf(approximate_area_20);
```

$$approximate\_area\_20 := \frac{1}{20} \sum_{i=0}^{19} e^{-\frac{1}{400} i^2}$$

$$approximate\_area\_20F := 0.7624738510 \qquad \textbf{(2.1.1.3)}$$

```
> approximate_area_200:=leftsum(f(x), x=0..1, 200);
  approximate_area_200F:=evalf(approximate_area_200);
```

$$approximate\_area\_200 := \frac{1}{200} \sum_{i=0}^{199} e^{-\frac{1}{40000} i^2}$$

$$approximate\_area\_200F := 0.7484029015 \quad (2.1.1.4)$$

Maple has a built in procedure for the numerical approximation of integrals, it employs a quadrature method and can be invoked by the command

evalf(Int(f(x), x=a..b));

```
> e1;
```

$$\int_0^1 e^{-x^2} \, dx \quad (2.1.1.5)$$

```
> professional_approximation_of_area:=evalf(e1);
```

$$professional\_approximation\_of\_area := 0.7468241328 \quad (2.1.1.6)$$

We can accept this number as the "exact" value of the integral, and use it to compute the error in the three previous computations.

```
> error_1:=abs(approximate_area_10F-
  professional_approximation_of_area);
```

$$error\_1 := 0.0309926913 \quad (2.1.1.7)$$

```
> error_2:=abs(approximate_area_20F-
  professional_approximation_of_area);
```

$$error\_2 := 0.0156497182 \quad (2.1.1.8)$$

```
> error_3:=abs(approximate_area_200F-
  professional_approximation_of_area);
```

$$error\_3 := 0.0015787687 \quad (2.1.1.9)$$

Rather than using the elementary left sum approach, we generally prefer to use slightly more advanced methods, known as the Trapezoidal rule and the Simpson's rule. Both will be discussed in class.

Using 20 sub-intervals, we now apply these methods to $\int_0^1 e^{-x^2} \, dx$.

```
> trap_20:=trapezoid(f(x), x=0..1, 20);
  trap_20F:=evalf(trap_20);
```

$$trap\_20 := \frac{1}{40} + \frac{1}{20} \sum_{i=1}^{19} e^{-\frac{1}{400} i^2} + \frac{1}{40} e^{-1}$$

$$trap\_20F := 0.7466708370 \quad (2.1.1.10)$$

```
> error_4:=abs(trap_20F-professional_approximation_of_area)
  ;
```

$$error\_4 := 0.0001532958 \quad (2.1.1.11)$$

Observe that the result of the Trapezoidal rule is much better than the result of a left sum with the same number of sub intervals. The Simpson's rule will do even better, as is shown below.

```
> simp_20:=simpson(f(x), x=0..1, 20);
  simp_20F:=evalf(simp_20);
```

$$simp\_20 := \frac{1}{60} + \frac{1}{60} e^{-1} + \frac{1}{15} \sum_{i=1}^{10} e^{-\left(\frac{1}{10} i - \frac{1}{20}\right)^2} + \frac{1}{30} \sum_{i=1}^{9} e^{-\frac{1}{100} i^2}$$

$$simp\_20F := 0.7468241839 \qquad (2.1.1.12)$$

```
> error_5:=abs(simp_20F-professional_approximation_of_area)
  ;
```

$$error\_5 := 5.11\ 10^{-8} \qquad (2.1.1.13)$$

```
>
```