Doing Numerical Integration in RStudio

T.Scofield

Definite integrals

For a built-in function like $\sin(x)$, integrating from 0 to π is as simple as

```
integrate(sin, 0, pi)
```

```
2 with absolute error < 2.2e-14
```

But, there may not *be* a built-in R function that suits your purposes. For instance, the pdf for exponential distributions is

$$f(x) := \begin{cases} 0, & \text{if } x < 0 \\ \lambda e^{-\lambda x}, & \text{if } x \geq 0 \end{cases}$$

implemented in R as `dexp()`. However, the integral one must calculate to find the mean involves the product of x with this function, and R does not have a built-in function for that.

When there is no built-in function, you have a couple of options for defining it yourself. One method is to use the native way R has for defining complicated user-defined functions:

```
meanIntegrand = function(x, lam) {
  return( (x>=0) * x * lam*exp(-lam*x) )
}
```

The other method is to use `makeFun()`, suitable when the function definition is less-complicated, as it is here, where I set $\lambda = 0.2$:

```
meanIntegrand <- makeFun( x*dexp(x, 0.2) ~ x )
```

Once defined, you can use `integrate()` to calculate definite integrals. It is even possible to use $(-\infty)$ or ∞ as endpoints of integration:

```
integrate(meanIntegrand, -Inf, Inf)
```

```
5 with absolute error < 3.8e-05
```

Indefinite integration (i.e., antidifferentiation)

The `antiD()` command is provided in the **mosaicCalc** package. Make sure you have it loaded before trying things such as the following.

The idea is to find an antiderivative for a given function, for purposes such as evaluating that antiderivative at various inputs, or displaying a graph of the antiderivative. Let's take $f(x) = \sin x$, a function we understand. We can produce an antiderivative, naming it $F(x)$, via

```
F <- antiD(sin(x) ~ x)     # There are many, with F(x) = -cos(x) + C
```

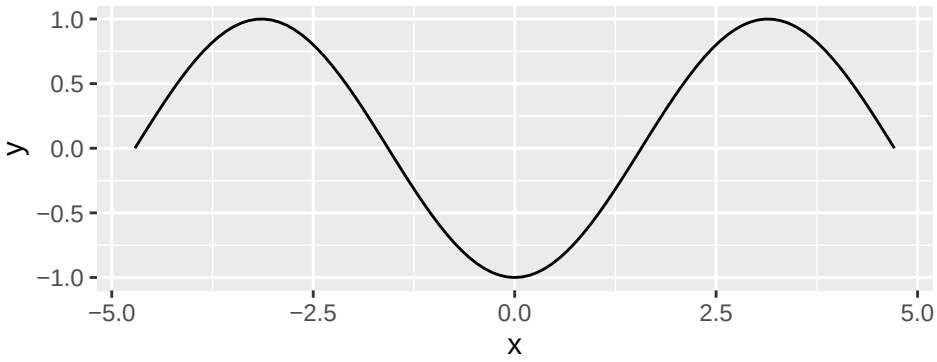
It can be evaluated:

```
F(3*pi/2) - F(0)
```

```
[1] 1
```

It can also be plotted:

```
gf_fun( F(x) ~ x, xlim = c(-3*pi/2, 3*pi/2))
```



On p. 133, Pruim introduces the function

$$f(x) = \begin{cases} x^2/9, & 0 \le x \le 3 \\ 0, & \text{otherwise} \end{cases}$$

which serves as a pdf because it is nonnegative and the total area under the curve is 1. We can produce its cdf and plot the two functions together:

```
require(gridExtra)     # package provides grid.arrange() for side-by-side plotting
```

```
Loading required package: gridExtra
```

```
Attaching package: 'gridExtra'
```

```
The following object is masked from 'package:dplyr':

    combine
```

```
myf <- makeFun( (x>=0 & x<=3) * x^2/9 ~ x )
myF <- antiD( myf(x) ~ x )
p1 <- plotFun(myf(x) ~ x, xlim = c(-1, 4))
p2 <- plotFun(myF(x) ~ x, xlim = c(-1, 4))
grid.arrange(p1, p2, nrow=1)
```