18. Symbolic and Automatic Differentiation

18.1 Symbolic Differentiation

Symbolic differentiation is the calculation of the derivative of a formula, very much according to rules you learned in calculus. The Maple command for symbolic differentiation is diff, and its "inert" version is Diff.

```
[> expsin := exp(sin(x));
```

The "inert" version of diff, the Diff, simply echoes the command, while diff executes:

```
[> Diff(expsin,x) = diff(expsin,x);
```

We can compute the second, third, fourth, ... derivative applying diff repeatedly, but doing this 25 times is cumbersome. Therefore, Maple has the dollar operator:

```
[> x$10;                                 # shortcut to build a sequence
[> diff(expsin,x$10);                    # here we have the 10-th derivative
```

In many applications, what we wish to derive is not defined explicitly, but implicitly by an equation. To recall what implicit differentiation is, we first do it the long way on the equation of a circle $x^2 + y^2 = 1$. The equation "circle" relates y to x. We first tell Maple we view y as a function of x:

```
[> alias(y=y(x)):
[> circle := x^2 + y^2 = 1;
```

To get diff(y,x), we differentiate the defining equation. Remember: the differentiation of an equation is an equation!

```
[> equ := diff(circle,x);
```

Now we need to solve for the derivative of y with respect to x:

```
[> solve(equ,diff(y,x));
```

To short way goes like this :

```
[> implicitdiff(x^2+z^2=1,z,x);
```

And we can do it over and over again:

```
[> implicitdiff(x^2+z^2=1,z,x$3);        # 3-rd derivative
```

18.2 Automatic Differentiation

Automatic differentiation is the calculation of the derivative of a function; the result is again a function. The Maple command for automatic differentiation is D. We need automatic differentiation for two reasons. First: not every function can be represented by a nice formula. Second, even if there is a formula, we may have to deal with huge expression swell which renders the result of symbolic differentiation very difficult to use.

First we illustrate the difference between diff and D. For the example expsin from above, we first make a function:

```
[> funexpsin := unapply(expsin,x);
[> derfunexpsin := D(funexpsin);
[> derfunexpsin(1.2);
```

It is instructive to look at the following commands:

```
[> diff(cos(t),t);                    # differentiation of the formula cos(t)
[> diff(cos,t);                       # wrong !
[> D(cos(t));                         # also wrong
[> D(cos);                            # differentiation of the cosine function
```

In dynamical systems we often wish to apply a function again and again. One example is the following:

```
[> julia := proc(z,n::nonnegint)
>    description `applies map z -> z^2 + 0.1 n times`:
>    local i,y:
>    y := z:
>    for i from 1 to n do
>      y := evalf(y^2 + 0.1):
>    end do:
>    RETURN(y):
[> end proc;
[> julia(1,2);
```

Since the argument of our procedure julia can be anything, we can as well give it a name to create a formula:

```
[> jul7 := julia(z,7):                # apply 7 times the function z^2 + 0.1
[> difjul7 := diff(jul7,z$9):         # differentiate it 9 times
[> fundifjul7 := unapply(difjul7,z): # turn the formula into a function
[> fundifjul7(0.2);
[> dfj9 := D[1$9](julia):             # differentiate function 9 times w.r.t. z
[> dfj9(0.2,7);
```

18.3 Assignments

1. Give the Maple commands to get the value of the 7-th derivative of $e^{x^{10}+2}$ at $x = 1$.
   Also give the value you obtained.
2. Give the Maple command(s) to compute $\frac{\partial^8 f}{\partial^5 x \partial^3 y}$ for $f(x, y) = e^{2x+\cos(y)}$.
3. Consider the curve defined by $f(x, y) = 3 + 2x + y + 2x^2 + 2xy + 3y^2 = 0$.
   Locally on the curve we can view y as a function of x, i.e.: $y = y(x)$.
   Compute formulas for the first and second derivative of y with respect to x.
4. Compute the derivative of the function $f(x) = \min(x^2 + 1, 2x + 3)$.