Differentiation

Table of Contents

Differentiation is the most important thing in calculus so let's get to it!

Using `diff`

What about calculus? Don't worry -- Matlab will not let you down! Suppose you'd like to differentiate the function `log(6*x+2)`. You could either do it yourself or... just ask Matlab to do it with the `diff` command:

```
syms x
diff(log(6 * x + 2))
```

```
ans =

6/(6*x + 2)
```

Higher Derivatives

What could be easier?

Would you like to find the *third* derivative of the function `log(6*x+2)`? That's easy too -- just pass 3 as a second parameter to the `diff` command:

```
diff(log(6 * x + 2), 3)
```

```
ans =

432/(6*x + 2)^3
```

A Different Variable

Suppose our expression has two variables and we want the derivative with respect to one of them. As usual `x` is the default

```
syms a x
diff(a^3*x^4)
```

```
ans =

4*a^3*x^3
```

but we can tell Matlab differently.

```
diff(a^3*x^4,a)
```

```
ans =

3*a^2*x^4
```

We can even do the second derivative with respect to `a`.

```
diff(a^3*x^4,a,2)
```

```
ans =

6*a*x^4
```

Wait, that Second Parameter?

Matlab is smart. If the second parameter is a variable it will take the derivative with respect to that variable. If it's a number it will take that numbered derivative. If it sees a variable and *then* a number it will take that numbered derivative with respect to that variable.

Plugging Stuff In - Using subs

It may seem a bit late but this is the perfect time to talk about plugging things into symbolic expressions. Here's how. Suppose we simply want to plug `x=3` into `x^2-x+2`. We do:

```
subs(x^2-x+2,x,3)
```

```
ans =

     8
```

So now to take the derivative and then plug in, we simply nest the commands. Here's the second derivative of `x^3+exp(x^2)` with `x=1` plugged in:

```
subs(diff(x^3+exp(x^2),2),x,1)
```

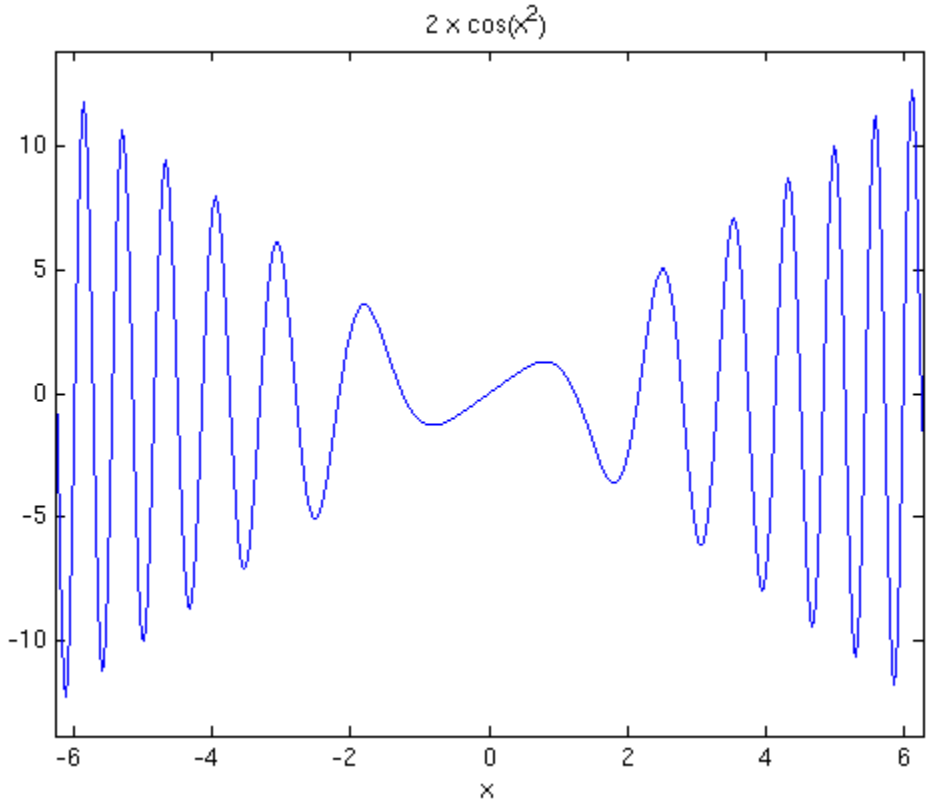
```
ans =

   22.3097
```

Plotting Derivatives

Likewise we can nest `diff` inside `ezplot`. Here's an example, a plot of the derivative of `sin(x^2)`:

```
ezplot(diff(sin(x^2)))
```



Published with MATLAB® 8.0