CSMMA16

Introduction to Calculus in R

Aim

The aim of this session is to briefly introduce the use of R in solving calculus problems.

Differentiation, Plotting and Newton Raphson

In this part of the practical we use R to produce derivatives of simple expressions and to describe features of the associated curves.

A Polynomial

First type

```
pol<-function(x) a*x^3 + b*x^2 + c*x + d
```

This constructs a cubic polynomial function in `x` named `pol`. Plot this function for a=3, b=4, c=-3, d=4 using the following (type `?seq`, or `?plot` to find out more about these functions):

```
a<-3; b<-4; c<--3; d<-4

x<-seq(-3,2,0.1); y<-pol(x)

plot(x,y,type="l")
```

R produces symbolic derivatives of simple expressions using the functions `D` and `deriv`.

`D` does univariate differentiation; `deriv` uses `D` to do multivariate differentiation. The output of `D` is an expression whereas the output of `deriv` can be an executable function.

To see this, try running the following:

```
Dx<-D(expression(a*x^3 + b*x^2 + c*x + d),"x")

Dx

dx<-deriv(( ~ a*x^3 + b*x^2+ c*x + d), c("x"), func=TRUE,
hessian=TRUE)

dx(0.28)
```

The `Deriv` function in the add-on package (library) Deriv also does symbolic differentiation of simple expressions and functions.

We will use the above expression object `dx` to determine the local maxima and minima of the polynomial $3x^3 + 4x^2 - 3x + 4$. We will also find the values of the polynomial at these points.

Newton-Raphson can be used to find values of x where the derivative is 0 (i.e. each root). First store the first and second derivatives in functions `grad` and `hess` as follows:

```
grad<-function(x)attr(dx(x),"gradient")[1]
```

```
hess<-function(x)attr(dx(x),"hessian")[1]
```

Then, starting with an initial guess for x, the next estimate can be found by

```
x=x-grad(x)/hess(x)
```

For each root, a good starting point can be found by looking at the plot. Use R to iterate onto the actual root (accurate to 4dp). The values of the polynomial at the max and min is produced by evaluating our function `pol` at these roots.

Compare your answers with the command

```
polyroot(c(c,2*b,3*a))
```

1

with a=3, b=4, c=-3.

An Exponential

Find the optimal value of the function $y=(1+2.5t)e^{-0.3t}$ and the point t.max at which this occurs using Newton-Raphson. Use R to iterate until t.max is found accurate to 4dp.

A suitable plot can be obtained using values of t in the interval (-1, 30). Look at the plot to find a suitable starting value of t. Also, try starting value t>7 to see what happens.

Compare your answer with the command

```
optimize(function(t) (1+2.5*t)*exp(-0.3*t), lower=-1,
upper=10,maximum = TRUE)
```

Assuming the derivative is the function grad.t, you can also try

```
uniroot(grad.t, lower=-1, upper=10)
```

Differential Equations

In this part we will solve first and second order differential equations analytically, use R to obtain numerical solutions and compare the results.

First Order

The output velocity of a motor can be described by the differential equation

$$2\frac{dO}{dt} + 4O = I.$$

Determine the complementary function (transient solution) and particular integral (steady state solution) for O when I is constant ($I = 1$) and hence find the particular solution if $O = 0$ at $t = 0$.

Next, use R to obtain a numerical solution and plot three graphs showing the theoretical solution, the numerical solution (simulated response) and the error between the theoretical and simulated responses for $t=(0,3)$.

Modify the differential equation so that $I = sin(5t)$ and repeat the above. Note that the particular integral is now of the form

$$O_{PI} = c\sin(5t + \emptyset), \text{ or alternatively, } O_{PI} = c_1 \sin(5t) + c_2\cos(5t)$$

Repeat the above for $\frac{dy}{dt} = -20y$, y=1, when t=0. Obtain the numerical solution using the option `method="euler"` and comment on the theoretical and numerical solutions.

Second Order

Overdamped Response

Consider a system with differential equation given by

$$\frac{d^2O}{dt^2} + 4\frac{dO}{dt} + 3O = 1.$$

Determine the transient and steady state solutions for O, and hence find the particular solution if O = 1 and dO/dt = 0 at t = 0.

Using R, obtain a numerical solution and plot graphs showing the theoretical and numerical solutions for t in the interval (0, 8). Comment on the plots.

Underdamped Response

Repeat the above for

$$2\frac{d^2O}{dt^2} + 4\frac{dO}{dt} + 3O = e^{-t}, \ O = 1 \text{ and } dO/dt = 0 \text{ at } t = 0.$$

Multivariable Example

Consider a function of two variables x_1 and x_2 given by

$$f(\boldsymbol{x}) = 2x_1^2 - x_1x_2 + 3x_2^2$$

Determine $\frac{df}{dx} = f_x$, the Hessian matrix f_{xx} and the eigenvalues of the Hessian matrix. Verify that (0,0) is a critical point and state whether it is a minimum, maximum or saddle point.

You can plot the function as follows:

```
x1<-seq(-3,3,length.out = 50); x2<-x1

fx1x2 <- function(x1,x2) 2*x1^2-x1*x2+3*x2^2

f.exp <- expression(f(x[1],x[2])==2*x[1]^2 - x[1]*x[2] +
3*x[2]^2)

y<- outer(x1,x2,bowl)
```

```
persp(x1,x2,y, theta = 30, phi = 30,expand = 0.5,col =
"lightblue",ltheta = 120,shade = 0.75,ticktype =
"detailed",xlab ="x1",ylab = "x2",zlab ="f(x1,x2)")
title(main =f.exp)
```

State whether or not the plot confirms your analysis.