Building an ODE model

- install package 'deSolve'
- load the package into memory using `library(deSolve)`

We will use a function called lsoda that is a versatile and robust numerical integrator for ordinary differential equations.

- powerful tool but you need to interact with it in a very specific way -- be careful with syntax!

Syntax for `lsoda`

Generic syntax for `lsoda` is:

```
output <- lsoda(init, tseq, ODEfunction, pars)
```

where:

`init` is the initial value of the state variable

`tseq` is a vector of the time points where the model will be evaluated

`ODEfunction` is a place-holder for the name of the function holding the model equations

`pars` is a vector containing any parameters used in the model

`output` is the variable where `lsoda` will return its results.

Syntax for `lsoda`

The function holding the model equations must have syntax:

```
ODEfunction <- function(tt, yy, pars) {
   derivs <- [insert model equations]
   return(list(derivs))
}
```

where:

`tt` is a variable used by R to keep track of the timestep

`yy` is the state variable (or vector of state variables, for multi-variate models)

`pars` is your vector of model parameters

`derivs` is an internal variable that records the time series of results

`lsoda` example: exponential growth

$$\frac{dN}{dt} = rN$$

```
expGrowthODE <- function(tt, NN, pars) {
   derivs <- pars['rr'] * NN
   return(list(derivs))
}
init <- 1
tseq <- seq(0, 20, by=0.01)
pars <- c(rr = 0.1)
```

Note new way of indexing a vector.
Access with `pars['rr']`
Don't need to use this method, but it's useful when you have lots of parameters.

Then call with:

```
expOutput <- lsoda( init, tseq, expGrowthODE, pars)
```

Output from `lsoda`

The output from our command:

```
expOutput <- lsoda( init, tseq, expGrowthODE, pars)
```

will be a matrix made up of two column vectors.

The first column will be the time points where the state of the system is recorded, and the second column will be the corresponding values of the state variable.

```
> expOutput
      time        1
 [1,] 0.00 1.000000
 [2,] 0.01 1.001001
 [3,] 0.02 1.002002
 [4,] 0.03 1.003005
...
```

So we can plot the dynamics with:

```
plot(expOutput[,1], expOutput[,2], col='blue', type='l')
```

`lsoda` example: logistic growth

$$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right)$$

```
logisticGrowthODE <- function(tt, NN, pars) {
   derivs <- pars['rr'] * NN * (1 - NN/pars['KK'])
   return(list(derivs))
}
init <- 1
tseq <- seq(0, 20, by=0.01)
pars <- c(rr = 0.1, KK = 100)
```

Then call with:

```
logisticOutput <- lsoda( init, tseq, expGrowthODE, pars)
```