

BCCS 2008/09: Graphical models and complex stochastic systems: Exercises 1

1. Learn **R**! Seriously, you can pick it up quickly as you go along. You might like to read the self-learn tutorial at <http://www.stats.bris.ac.uk/~peter/R/tutorial.pdf>, or one of the many documents under **Contributed** or possibly **Manuals** at <http://www.stats.bris.ac.uk/R/> (beware of the fact that some of these documents may refer to older versions of **R**, or to different platforms). Alternatively simply go for it by trying the practice session below.

Note that **R** is free, and you can download a copy for your own Windows/Mac/Linux PC at <http://www.stats.bris.ac.uk/R/>.

You start **R** on the BCCS PC's by looking under **Start | All Programs | Engineering Apps**. Type your input at the `>` prompt in the R console window.

```
2*3
z<-2*3
z
a<-c(1,7,4,5)
a[2]
a[2:4]
a[c(3,3,2)]
a[-2]
b<-1:4
a+b
a*b
a%*%b
m<-matrix(1:12,3,4)
m
m[2,c(1,3)]
nrow(m)
m^2
log(m)
m%*%a
if(z>5) print('big') else print('small')
for(i in 1:10) {
  print(c(i,i^2))
}

sort(a)
sum(a)/length(a)
sample(17)
sample(10,5)
rnorm(20)
z<-rnorm(20)
mean(z)
var(z)
median(z)
stem(rnorm(20))
x<-6*pi*(0:200)/200
plot(x,exp(-0.09*x)*sin(x))
plot(cumsum(runif(200))/(1:200))
x<-1:10
y<-3.4+1.2*x+rnorm(10)
plot(x,y)
abline(3.4,1.2,lty=2)
abline(lsfite(x,y))
hist(rgamma(500,2))
```

If you do not understand the output, experiment by changing some of the numbers. For help with a function, e.g. `cumsum`, simply type e.g. `?cumsum`.

As soon as you get beyond things you can achieve in one line, it is probably easiest to use a text editor to create code, and then copy-paste it into the R console window. Here is how to define and use a simple function:

```
plus<-function(a,b=1)
{
  a+b
}
plus(3,4)
plus(6)
plus(1:5)
```

2. **Random walks.** Use **R** to simulate a random walk, and plot its path as a graph.

A random walk is a stochastic process (sequence of random variables) S_n , $n = 0, 1, 2, \dots, N$ where S_0 is arbitrary and fixed, and $S_n = S_{n-1} + X_n$ where the steps $\{X_n\}$ are independent and identically distributed random variables.

Try this for normally (Gaussian) distributed steps (hint: `?rnorm`), and various run lengths N . What happens qualitatively as N gets large?

Try replacing `rnorm` by `rcauchy`.

Repeat, but plotting the cumulative average S_n/n , instead of S_n , against n . What have you learnt?

3. **Percolation.** Draw a $N \times N$ grid of squares (hint: you may want to use `plot` and `lines`). Now do it again, but where each line segment connecting adjacent grid nodes is either plotted or not plotted with probabilities p and $1 - p$, independently for each segment. This is a type of random graph: this problem is called ‘bond percolation on a square lattice’, and might be considered a simple model for the flow of liquid through porous rock.

Notice that the qualitative appearance of the plot varies with p ; in particular look carefully at the ‘cluster’ of nodes that are connected to the node at the middle of the grid. (It may help to print your random graph and draw round the edge of this cluster. Better still but *much* more intricate, change your code so as to plot the nodes that are part of this cluster in a contrasting colour (hint: `col='red'` as an argument to `points`)).

Can you formulate a conjecture about the dependence of the size of the cluster on p ?

4. If you have the time and interest, explore. The home page is <http://www.r-project.org/index.html>. See the extensive list of downloadable add-on Packages at <http://www.stats.bris.ac.uk/R/>. Try programming something harder.