

Some Examples of Graphics in the Statistical Language R

M. Hlynka
Dept. of Mathematics and Statistics
University of Windsor

June 13, 2007.

1. Introduction.

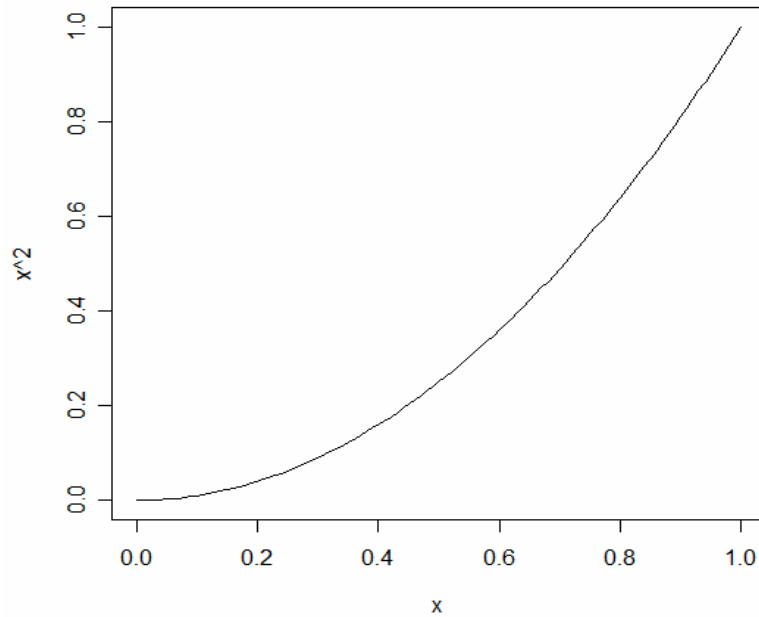
The statistical language R has some excellent graphics capabilities.

Unfortunately, there are many parameters and options and documentation is not always the best. One way to help with this situation is to have an example bank. If one can find an example close to what is needed, then the example can be modified. Everyone has specific needs. The examples presented here illustrate my needs and the needs of those who have asked me how to perform certain tasks. My examples can be combined with those of others to get a larger picture. The examples can be copied into R directly to see the commands in action. Changes are encouraged.

EXAMPLE 1: We begin with a graph of $y=x^2$.

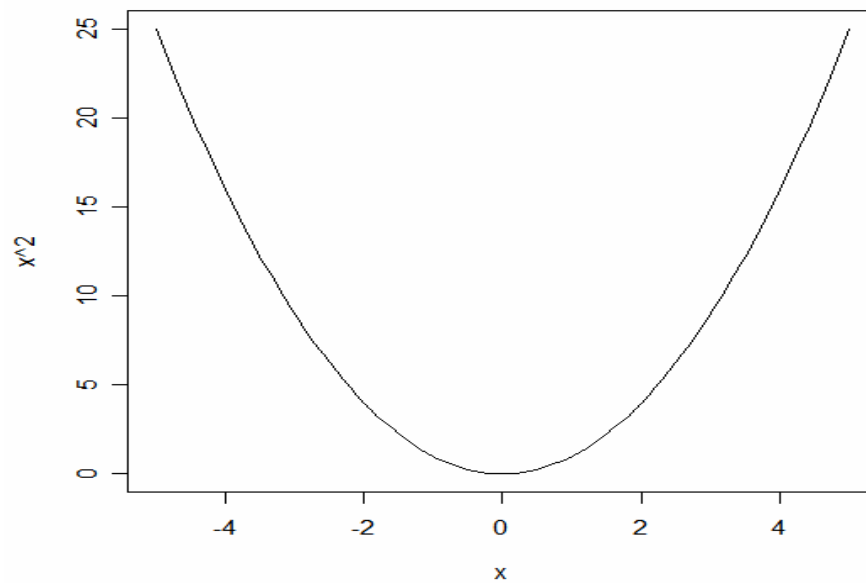
```
curve(x^2)
```

#The graphics system in R chooses its own scaling. I would not choose these limits.



```
curve(x^2,-5,5)
```

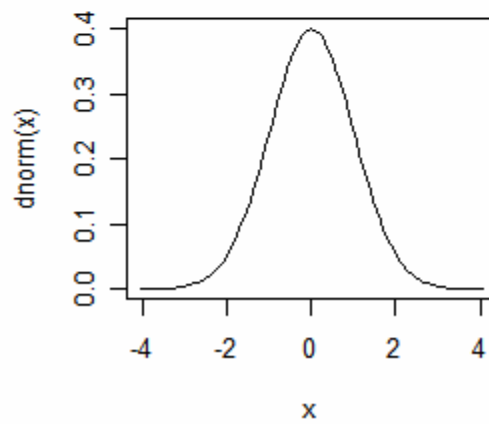
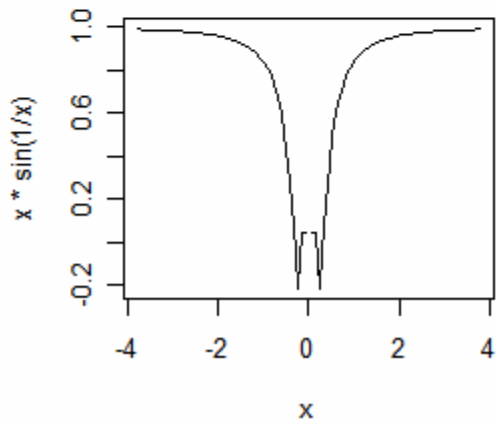
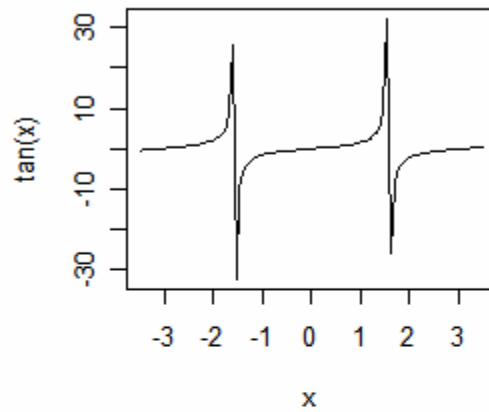
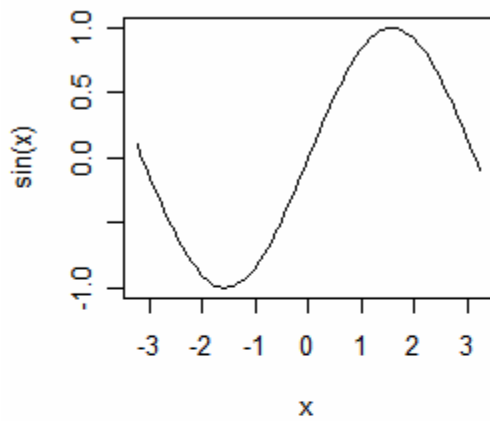
#We control the horizontal axis from $x=-5$ to $x=5$. That's better.



EXAMPLE 2

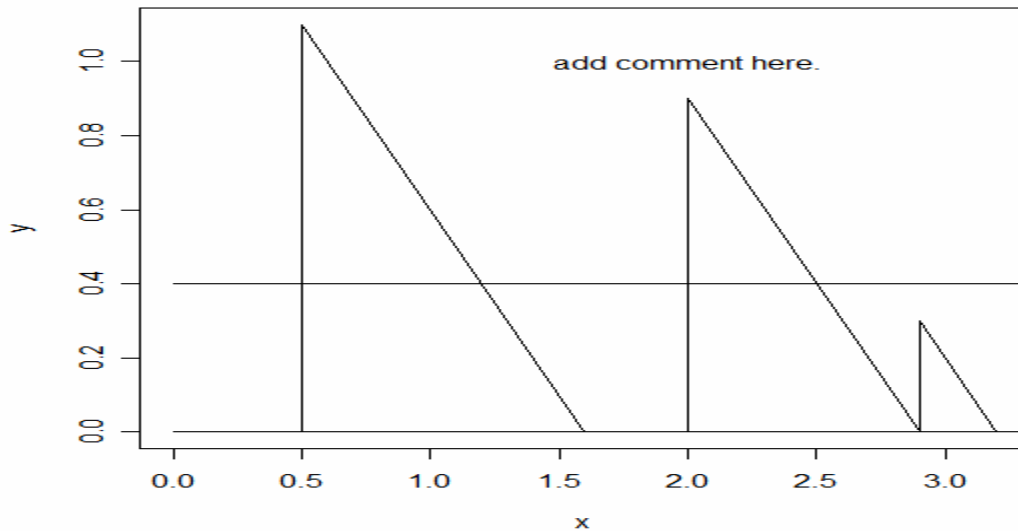
```
op= par(mfrow=c(2,2)) #op means options; par means parameters  
curve(sin(x))  
curve(tan(x))  
curve(x*sin(1/x))  
curve(dnorm(x))
```

This will give four (2 x 2) small plots



EXAMPLE 3: My colleague needs diagrams like this one.

```
x=c(0.0, 0.5, 0.5, 1.6, 2.0, 2.0, 2.9, 2.9, 3.2)
y=c(0.0, 0.0, 1.1, 0.0, 0.0, 0.9, 0.0, 0.3, 0.0)
# The x,y values were actually entered in spreadsheet fashion using the command
# data.entry(x,y), not as shown above.
x1=c(0.0, 3.3) #Add level x line
y1=c(0.4, 0.4)
x2=c(0.0, 3.3) #Add line for x axis
y2=c(0, 0)
plot(x,y,t="l") #The "l" refers to "line"
lines(x1,y1,t="l") #the plot command has to come first.
lines(x2,y2,t="l")
text(2,1,"add comment here.") # It is centered at point (2,1).
```



EXAMPLE 4: I need this to draw random walks.

```
#Again use data.entry(x,y)
```

```
#This sets up a spread sheet to aid for data entry. The values chosen for x and y are
```

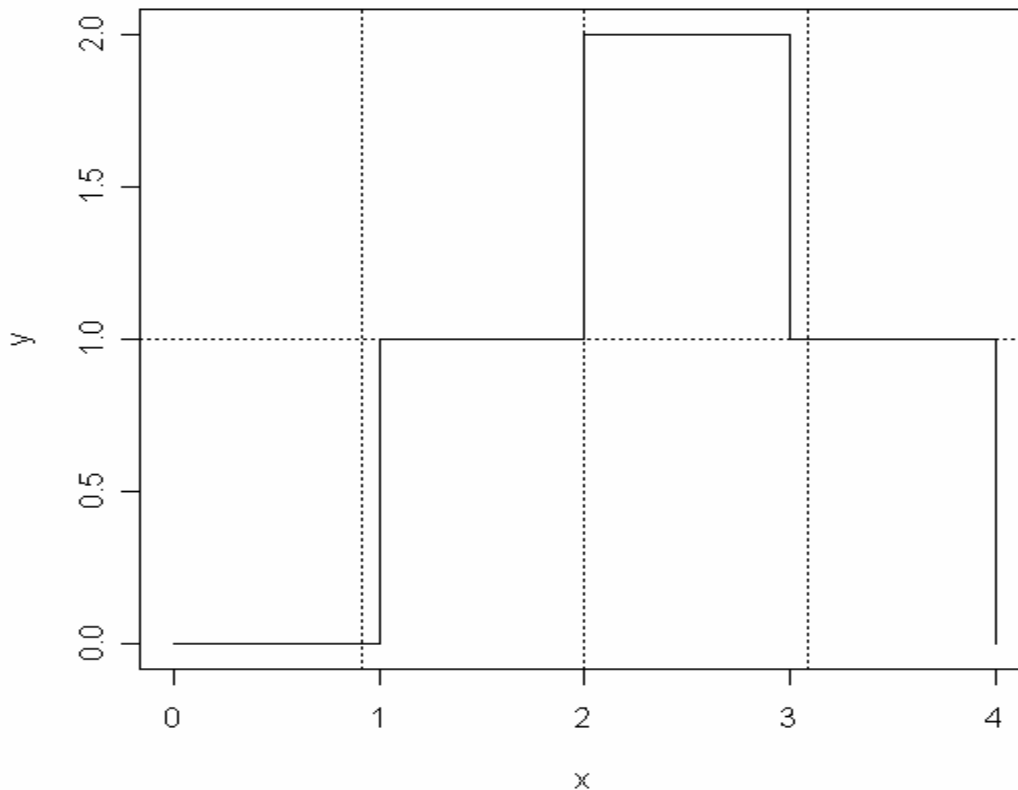
```
x=c(0, 1, 1, 2, 2, 3, 3, 4, 4)
```

```
y= c(0, 0, 1, 1, 2, 2, 1, 1, 0)
```

```
plot(x,y,"l") # The (x,y) points are plotted with the joining lines ("l")
```

```
grid(4,2, col="black") # This gives 4 vertical and 2 horizontal sections.
```

This will plot a random walk in the plane. Then add grid lines. The default color is light grey which is hard to see. So I made the color "black."



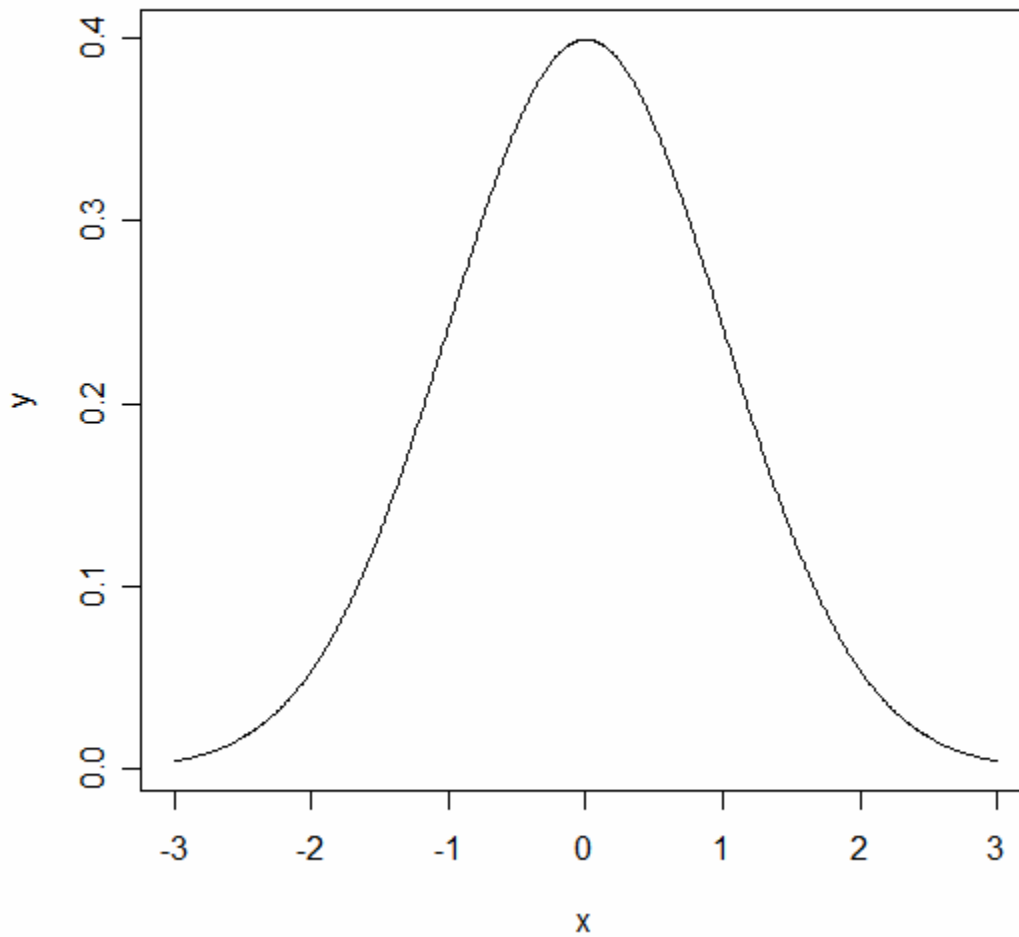
EXAMPLE 5:

I was asked how statisticians get diagrams of pdf's. This plots a normal pdf. We can do the same for t, F, chi-square, etc.

```
x=seq(-3,3,.01) # This gives lots of values on the x axis
```

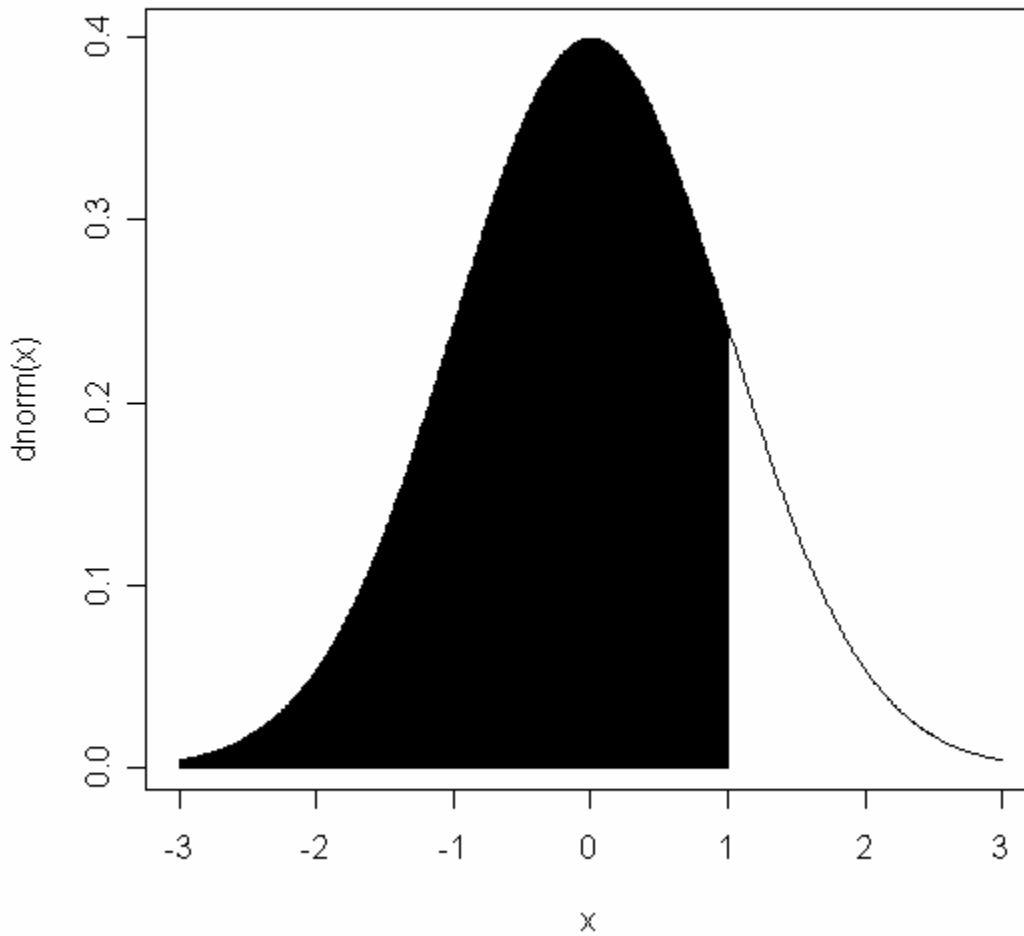
```
y=dnorm(x) #dnorm is the pdf of a normal(0,1)
```

```
plot(x,y,"1")
```



EXAMPLE 6. The following code shows an overlay of graphs. A statistics book author wanted a small diagram to put over the normal tables that were computed.

```
>x=seq(-3,3,.01)
> plot(x,dnorm(x),"l")
> x=seq(-3,1,.01)
> lines(x,dnorm(x),"h")
The "h" part is the blackened region.
```



EXAMPLE 7: This command gives information about some plotting commands/

```
library(help="graphics")
```

The output is:

Information on package 'graphics'

Description:

```
Package:      graphics
Version:     2.4.1
Priority:     base
Title:       The R Graphics Package
Author:      R Development Core Team and contributors worldwide
Maintainer:  R Core Team <R-core@r-project.org>
Description: R functions for base graphics
Depends:     grDevices
License:     GPL Version 2 or later.
Built:      R 2.4.1; ; 2006-12-18 09:43:18; windows
```

Index:

```
Axis          Generic function to add an Axis to a Plot
abline        Add a Straight Line to a Plot
arrows        Add Arrows to a Plot
assocplot     Association Plots
axTicks       Compute Axis Tickmark Locations
axis          Add an Axis to a Plot
axis.POSIXct  Date and Date-time Plotting Functions
barplot       Bar Plots
box           Draw a Box around a Plot
boxplot       Box Plots
bxp           Draw Box Plots from Summaries
cdplot        Conditional Density Plots
contour       Display Contours
coplot        Conditioning Plots
curve         Draw Function Plots
dotchart      Cleveland Dot Plots
filled.contour Level (Contour) Plots
fourfoldplot  Fourfold Plots
frame         Create / Start a New Plot Frame
graphics-package The R Graphics Package
grid          Add Grid to a Plot
hist          Histograms
hist.POSIXt   Histogram of a Date or Date-Time Object
```

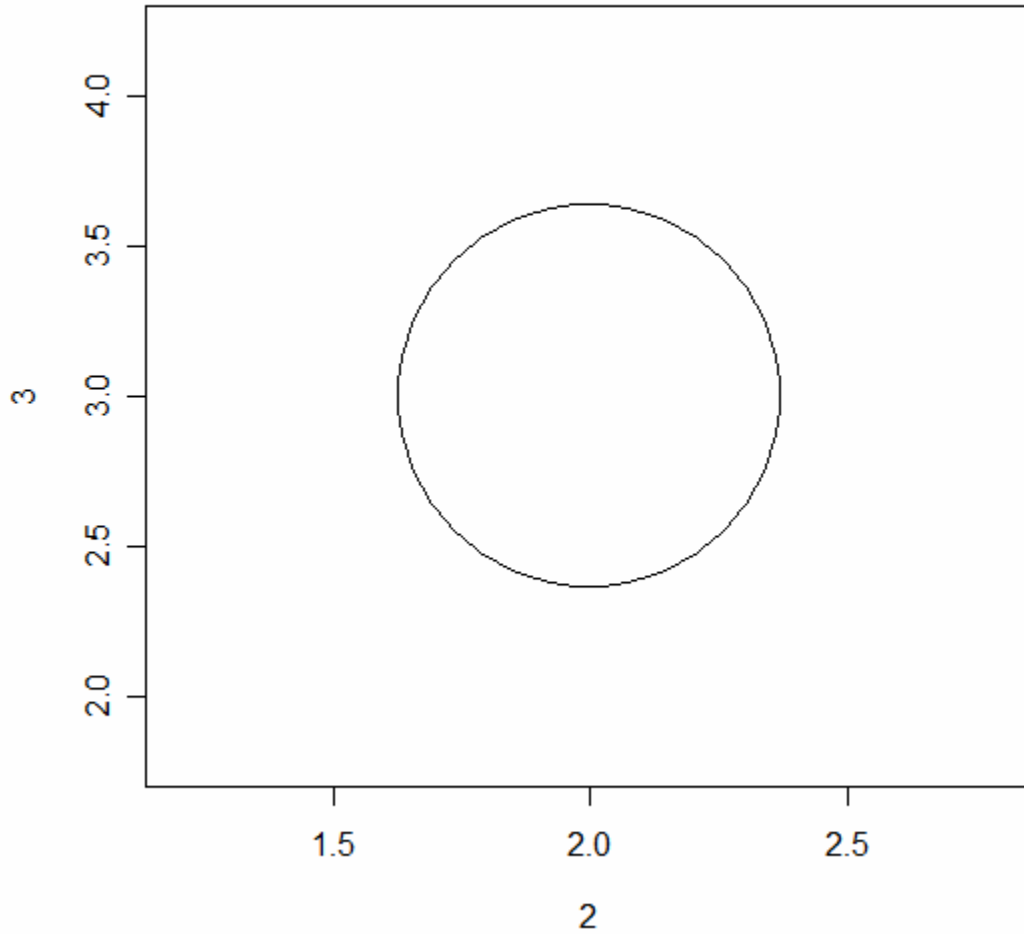
identify	Identify Points in a Scatter Plot
image	Display a Color Image
layout	Specifying Complex Plot Arrangements
legend	Add Legends to Plots
lines	Add Connected Line Segments to a Plot
locator	Graphical Input
matplot	Plot Columns of Matrices
mosaicplot	Mosaic Plots
mtext	Write Text into the Margins of a Plot
pairs	Scatterplot Matrices
panel.smooth	Simple Panel Plot
par	Set or Query Graphical Parameters
persp	Perspective Plots
pie	Pie Charts
plot	Generic X-Y Plotting
plot.data.frame	Plot Method for Data Frames
plot.default	The Default Scatterplot Function
plot.design	Plot Univariate Effects of a 'Design' or Model
plot.factor	Plotting Factor Variables
plot.formula	Formula Notation for Scatterplots
plot.histogram	Plot Histograms
plot.table	Plot Methods for 'table' Objects
plot.window	Set up World Coordinates for Graphics Window
plot.xy	Basic Internal Plot Function
points	Add Points to a Plot
polygon	Polygon Drawing
rect	Draw One or More Rectangles
rug	Add a Rug to a Plot
screen	Creating and Controlling Multiple Screens on a Single Device
segments	Add Line Segments to a Plot
spineplot	Spine Plots and Spinograms
stars	Star (Spider/Radar) Plots and Segment Diagrams
stem	Stem-and-Leaf Plots
stripchart	1-D Scatter Plots
strwidth	Plotting Dimensions of Character Strings and Math Expressions
sunflowerplot	Produce a Sunflower Scatter Plot
symbols	Draw Symbols (Circles, Squares, Stars, Thermometers, Boxplots) on a Plot
text	Add Text to a Plot
title	Plot Annotation
xinch	Graphical Units

EXAMPLE 8: Draw a circle.

```
symbols(2,3,circles=c(1))
```

The (2,3) gives the center of the circle and the 1 is the radius.

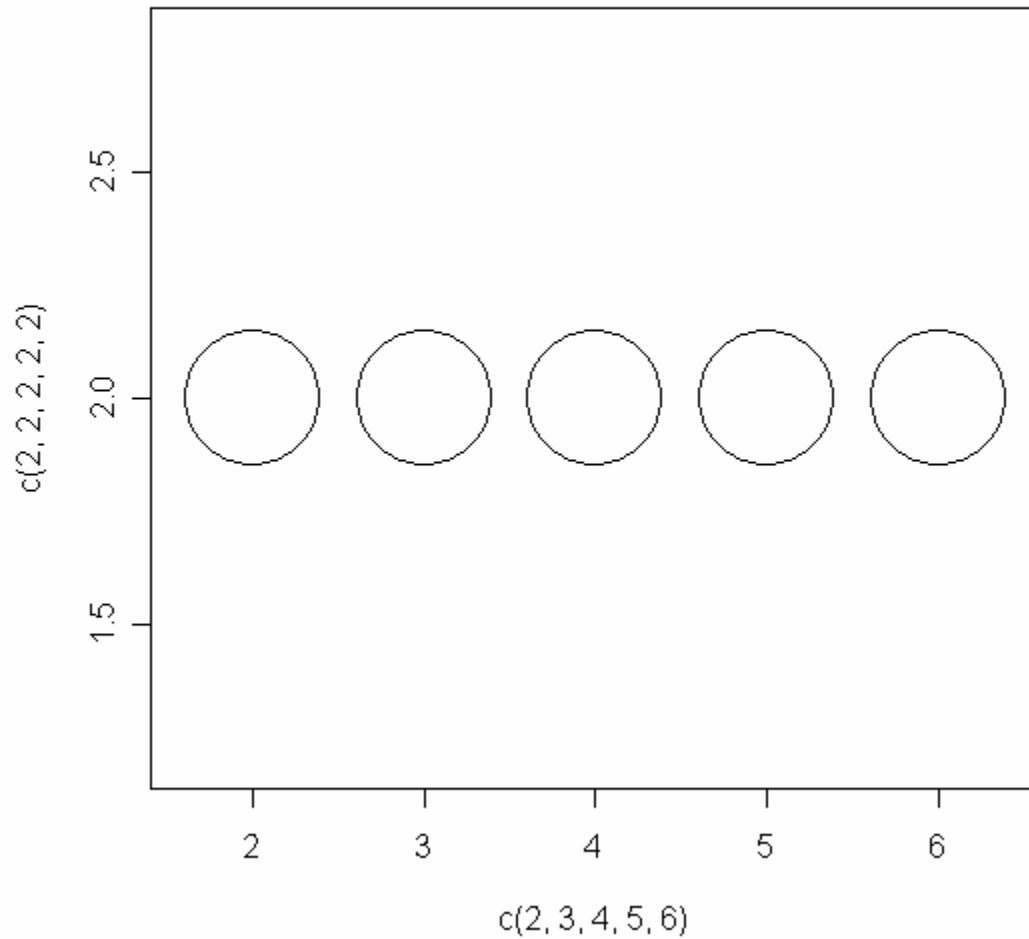
Output is



EXAMPLE 9: Draw 5 circles.

`symbols(c(2,3,4,5,6),c(2,2,2,2,2),circles=c(.4,.4,.4,.4,.4), inches=FALSE)`

This gives 5 circles centered at (2,2), (3,2), (4,2) (5,2) (6,2) with radii all equal to 0.4. The `inches=FALSE` is important because the default will change all the .4 to 1.

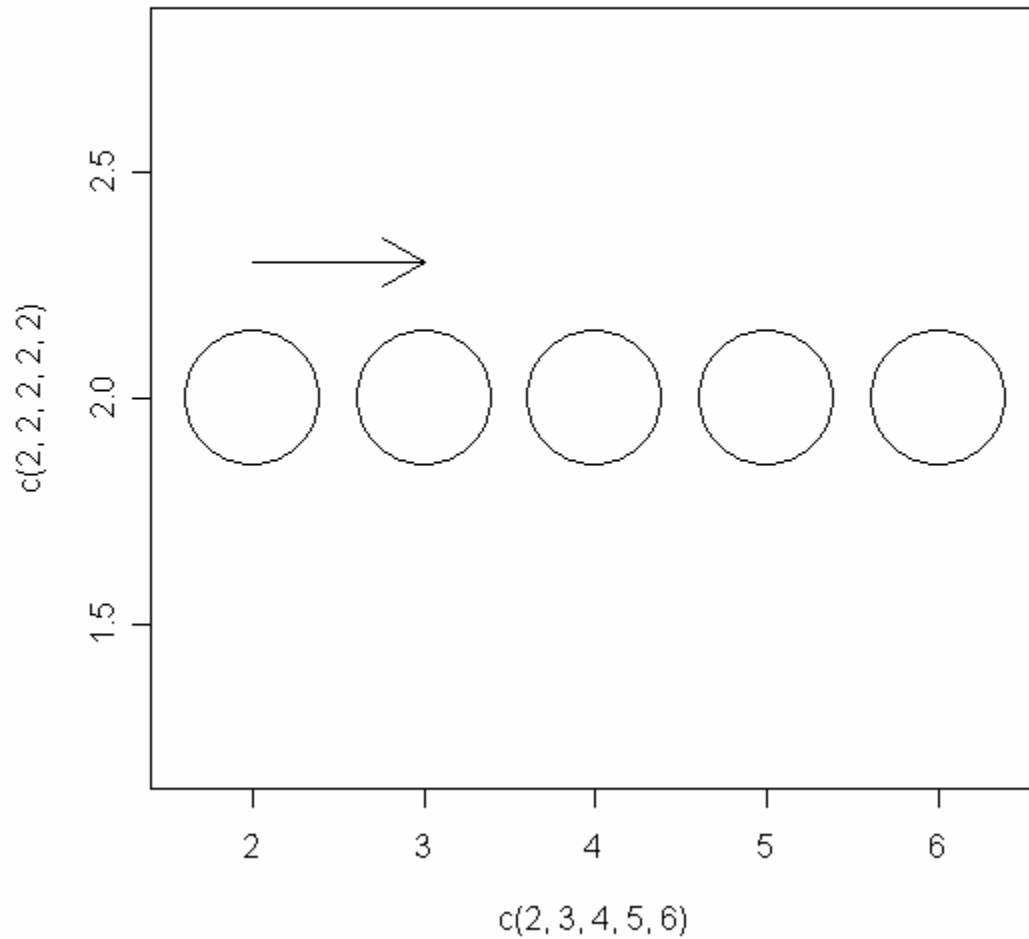


EXAMPLE 10:

```
symbols(c(2,3,4,5,6),c(2,2,2,2,2),circles=c(.4,.4,.4,.4,.4), inches=FALSE)
```

```
arrows(2,2.3,3,2.3)
```

#This adds an arrow from (2,2.3) to (3,2.3).

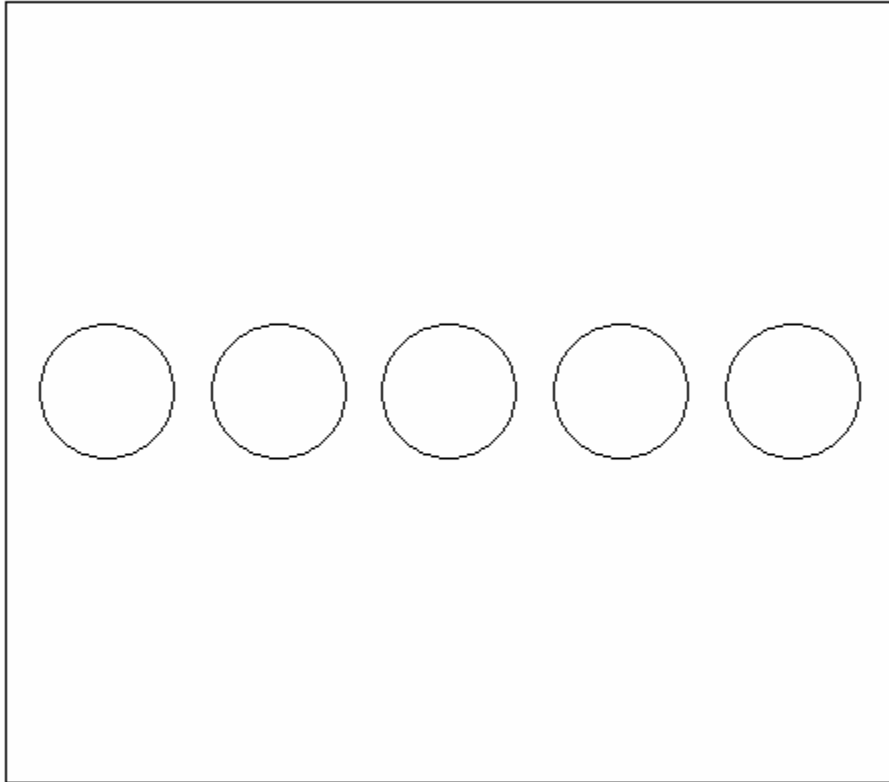


EXAMPLE 11: No axis labels this time.

```
symbols(c(2,3,4,5,6),c(2,2,2,2,2),circles=c(.4,.4,.4,.4,.4),  
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
```

#xaxt="n" means no tick marks on the x axis

#ann=FALSE means no labels on the axes



EXAMPLE 12: Include lots of arrows.

```
symbols(c(2,3,4,5,6),c(2,2,2,2,2),circles=c(.4,.4,.4,.4,.4),  
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
```

```
arrows(2.1,2.2,2.8,2.2)
```

```
arrows(3.1,2.2,3.8,2.2)
```

```
arrows(4.1,2.2,4.8,2.2)
```

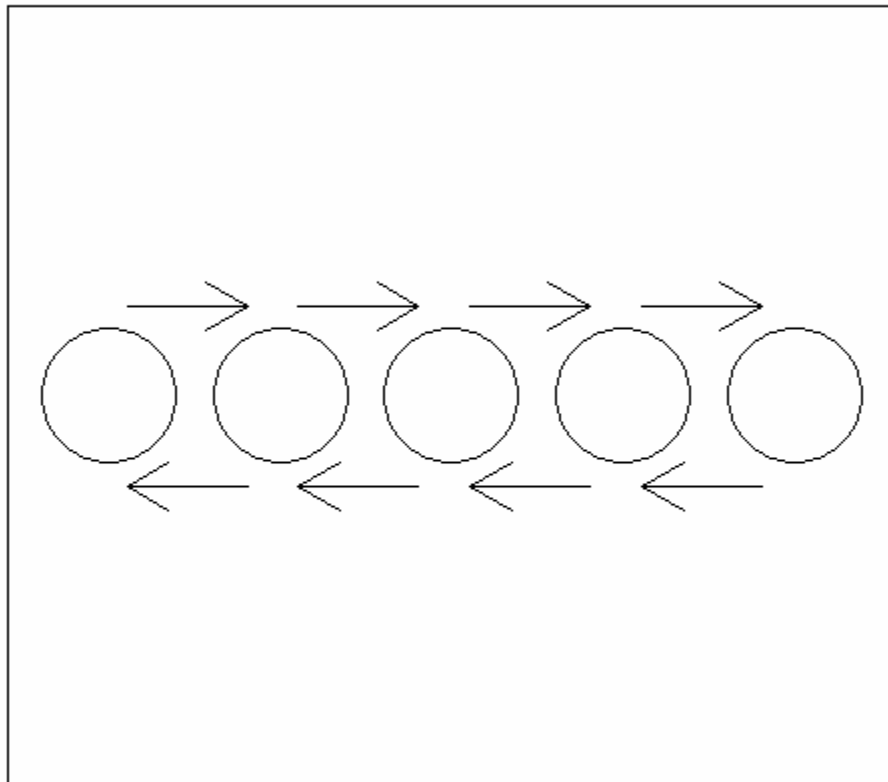
```
arrows(5.1,2.2,5.8,2.2)
```

```
arrows(5.8,1.8,5.1,1.8)
```

```
arrows(4.8,1.8,4.1,1.8)
```

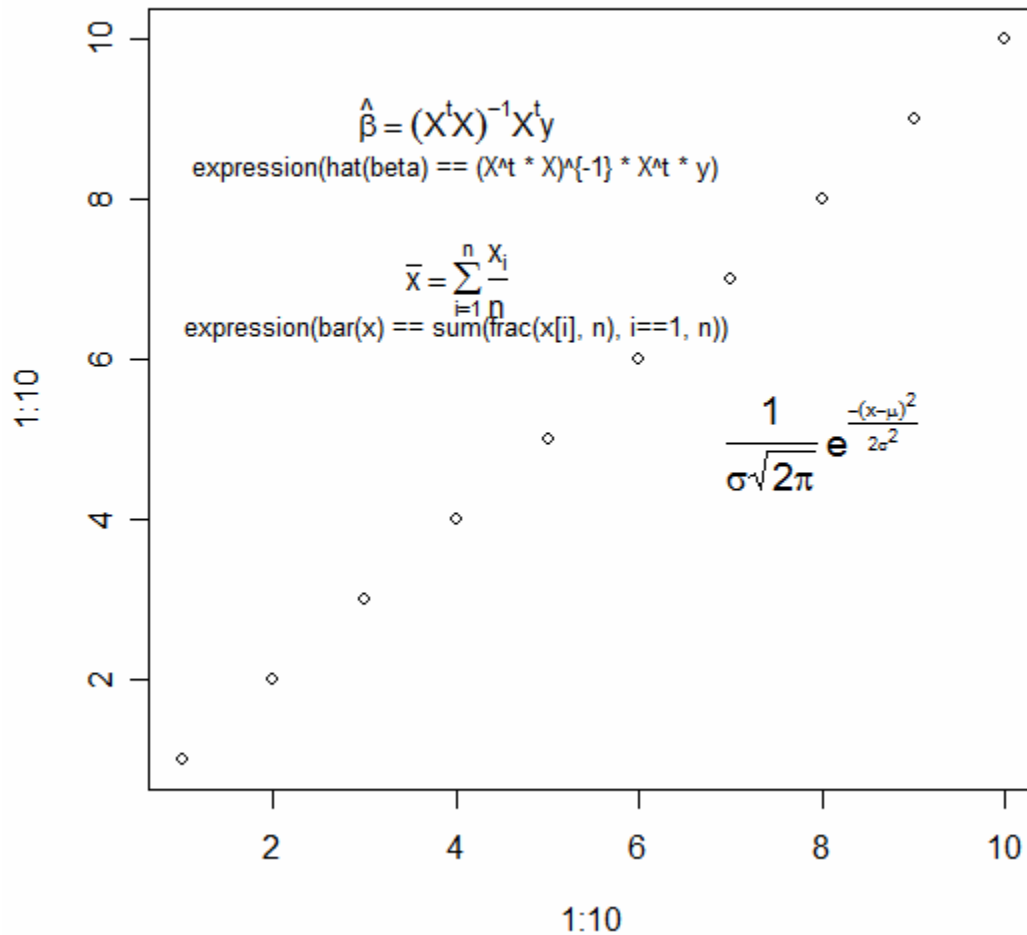
```
arrows(3.8,1.8,3.1,1.8)
```

```
arrows(2.8,1.8,2.1,1.8)
```



EXAMPLE 13: Here is an example from R's help menu about adding text to a plot.

```
plot(1:10, 1:10)
text(4, 9, expression(hat(beta) == (X^t * X)^{-1} * X^t * y))
text(4, 8.4, "expression(hat(beta) == (X^t * X)^{-1} * X^t * y)", cex = .8)
text(4, 7, expression(bar(x) == sum(frac(x[i], n), i==1, n)))
text(4, 6.4, "expression(bar(x) == sum(frac(x[i], n), i==1, n))", cex = .8)
text(8, 5, expression(paste(frac(1, sigma*sqrt(2*pi)), " ",
                             plain(e)^{frac(-(x-mu)^2, 2*sigma^2)})), cex = 1.2)
# cex stands for numeric character expansion factor which controls character size
```



es(x,y)

lin

EXAMPLE 14: Back to our flow chart. Add Greek letters with

symbols(c(2,3,4,5,6),c(2,2,2,2,2),circles=c(.4,.4,.4,.4,.4),

inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)

arrows(2.1,2.2,2.8,2.2) ; arrows(3.1,2.2,3.8,2.2)

arrows(4.1,2.2,4.8,2.2) ; arrows(5.1,2.2,5.8,2.2)

arrows(5.8,1.8,5.1,1.8) ; arrows(4.8,1.8,4.1,1.8)

arrows(3.8,1.8,3.1,1.8) ; arrows(2.8,1.8,2.1,1.8)

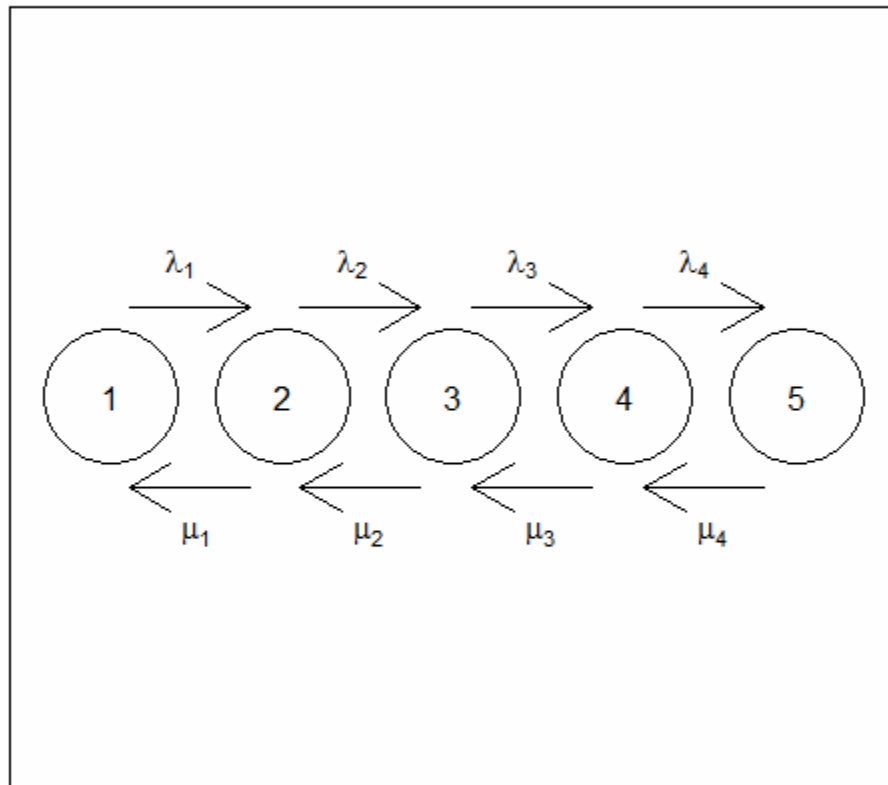
text(2.4,2.3, expression(lambda[1])); text(3.4,2.3, expression(lambda[2]))

text(4.4,2.3, expression(lambda[3])); text(5.4,2.3, expression(lambda[4]))

text(2.5,1.7, expression(mu[1])); text(3.5,1.7, expression(mu[2]))

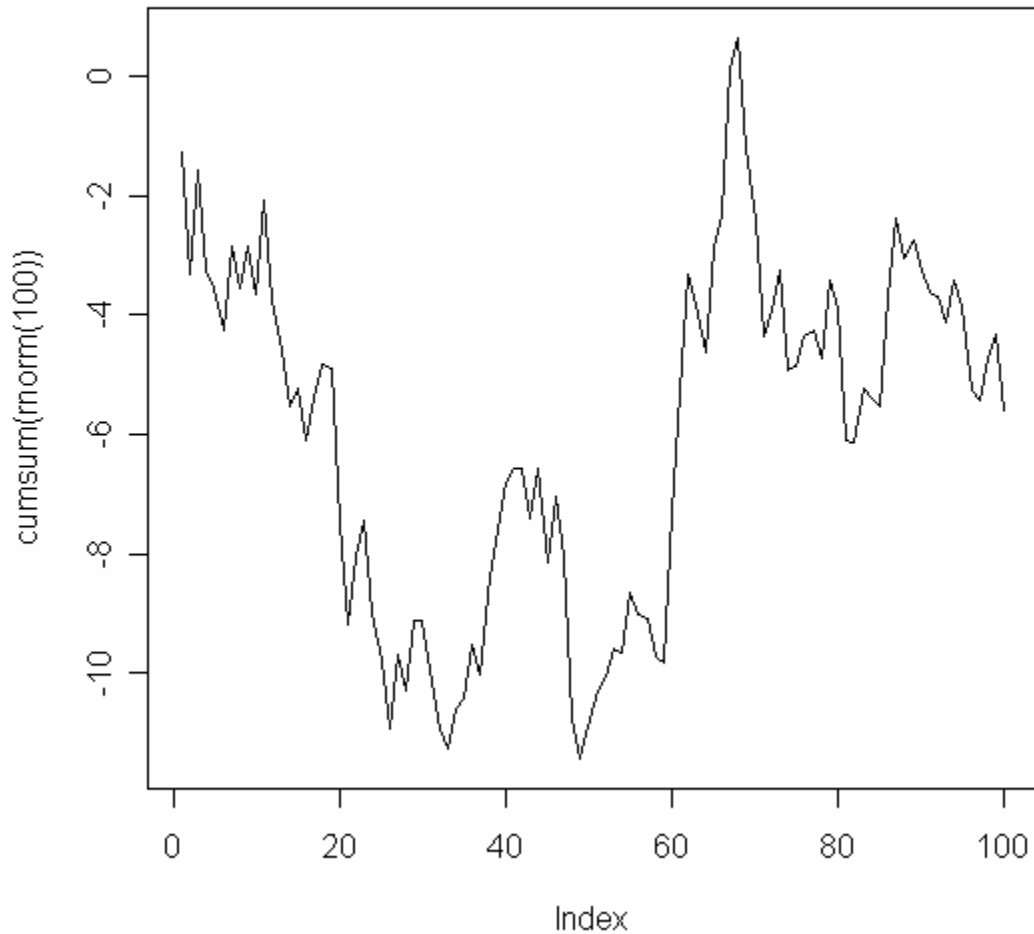
text(4.5,1.7, expression(mu[3])); text(5.5,1.7, expression(mu[4]))

text(2,2,1); text(3,2,2); text(4,2,3); text(5,2,4); text(6,2,5)



EXAMPLE 15:

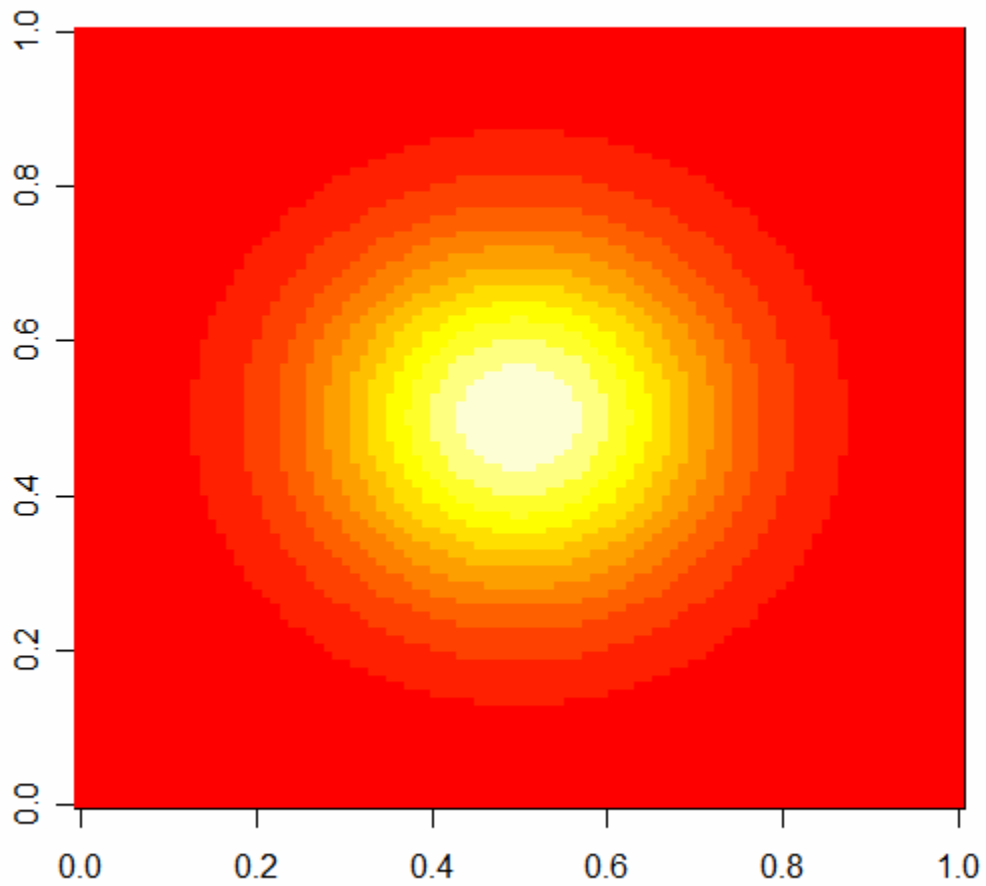
```
plot(cumsum(rnorm(100)),type="l")  
# This will plot a Brownian motion based on 100 normal values and their cumulative  
sums.
```



If you want the diagram to start from 0, use
`plot(c(0,cumsum(rnorm(100))),type="l")`

EXAMPLE 16: This is our attempt to plot a bivariate normal in colour, viewed from above.

```
a=seq(-3,3,l=100)
A=matrix(c(0),100,100) # create a 100x100 matrix full of zeros
for(i in 1:100){ for(j in 1:100){A[i,j]=dnorm(a[i])*dnorm(a[j])} }
image(t(A)[ncol(A):1,])
```

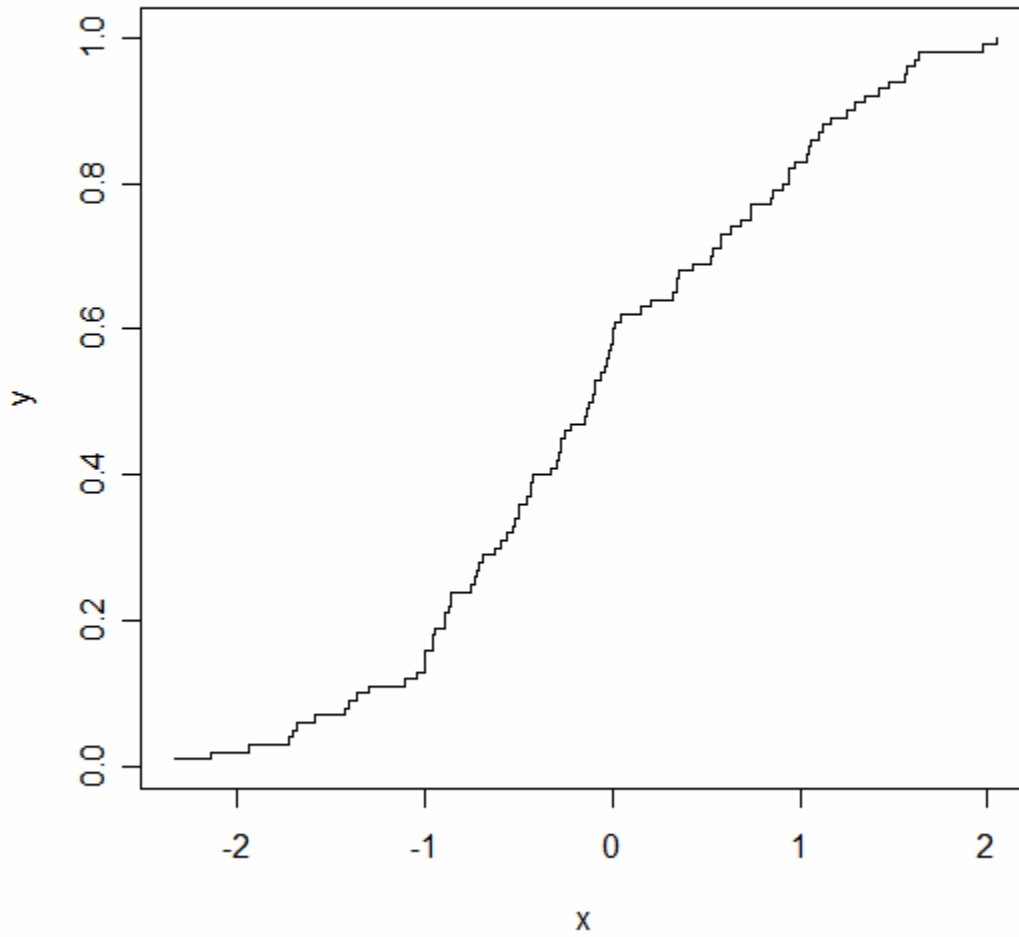


EXAMPLE 17: Empirical Distribution Function.

```
x=sort(rnorm(100))
```

```
y=seq(1:100)/100
```

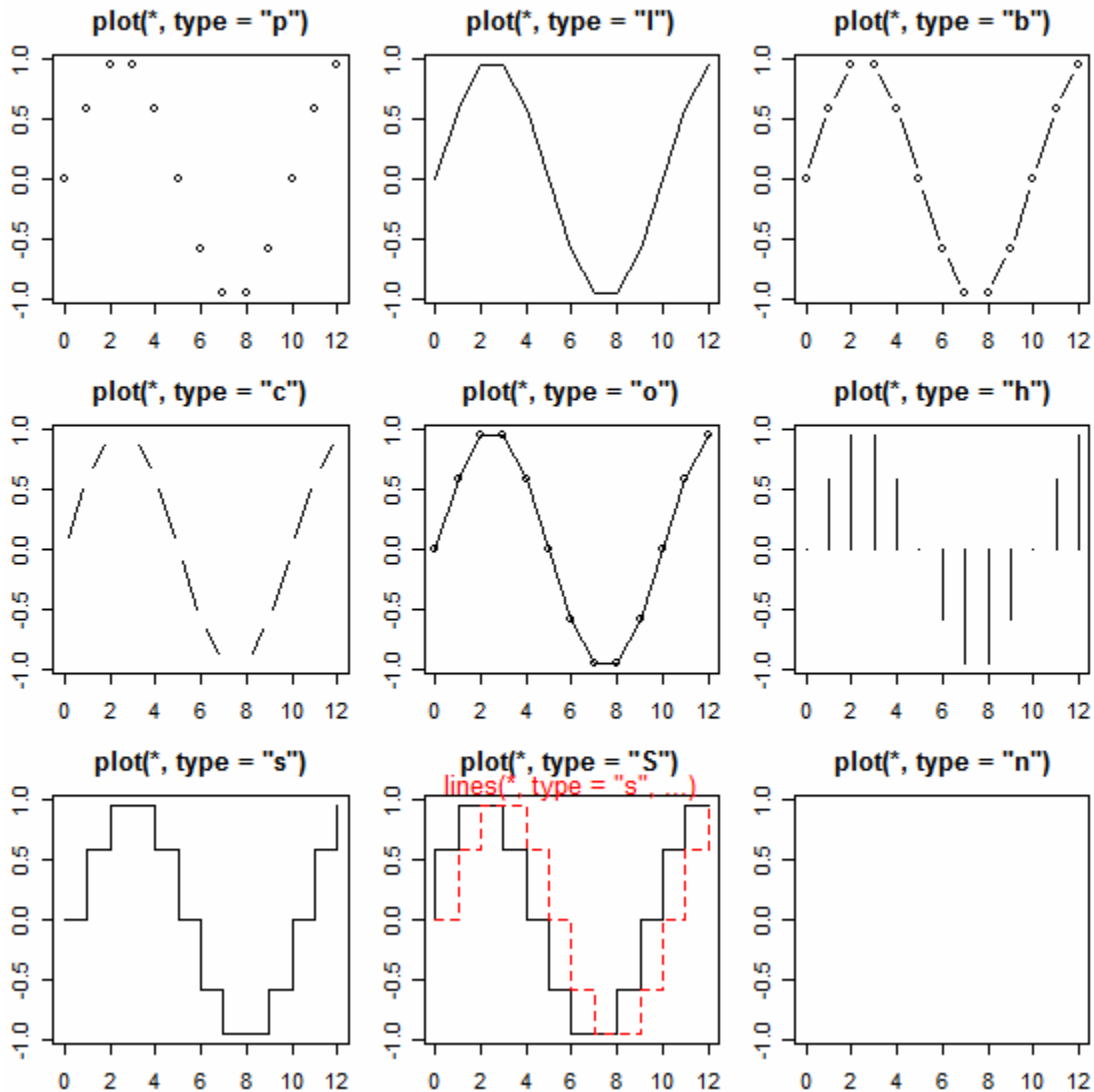
```
plot(x,y,"s")
```



EXAMPLE 18.

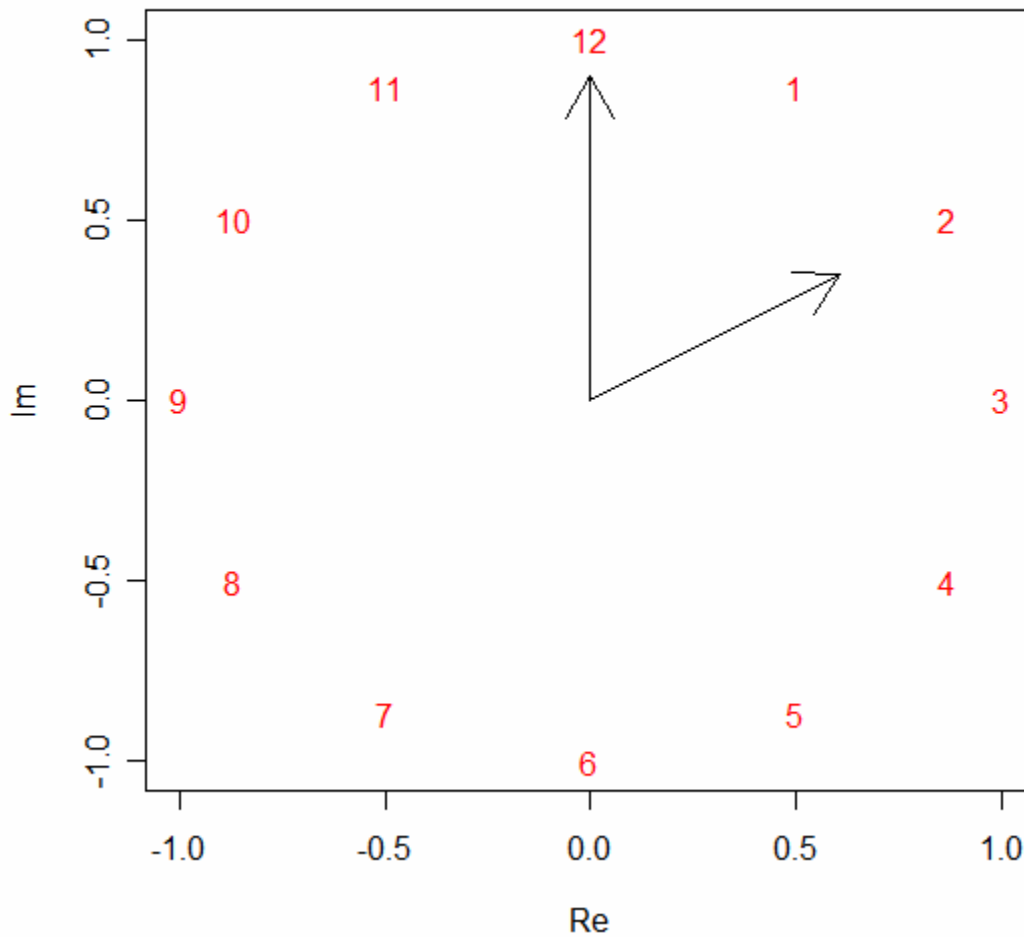
Examine different plot types.

```
## Show the different plot types
x <- 0:12
y <- sin(pi/5 * x)
op <- par(mfrow = c(3,3), mar = .1+ c(2,2,3,1))
for (tp in c("p","l","b", "c","o","h", "s","S","n")) {
  plot(y ~ x, type = tp,
       main = paste("plot(*, type = \"",tp,"\"",sep=""))
  if(tp == "S") {
    lines(x,y, type = "s", col = "red", lty = 2)
    mtext("lines(*, type = \"s\", ...)", col = "red", cex=.8)
  }
}
par(op) #This just resets the output size.
```



EXAMPLE 19 : This is my attempt at a clock. We are counting counterclockwise from the x axis so that is why the vector of numbers counts 2,1,12,11,10,9,...,3. I can omit the labels if I wish.

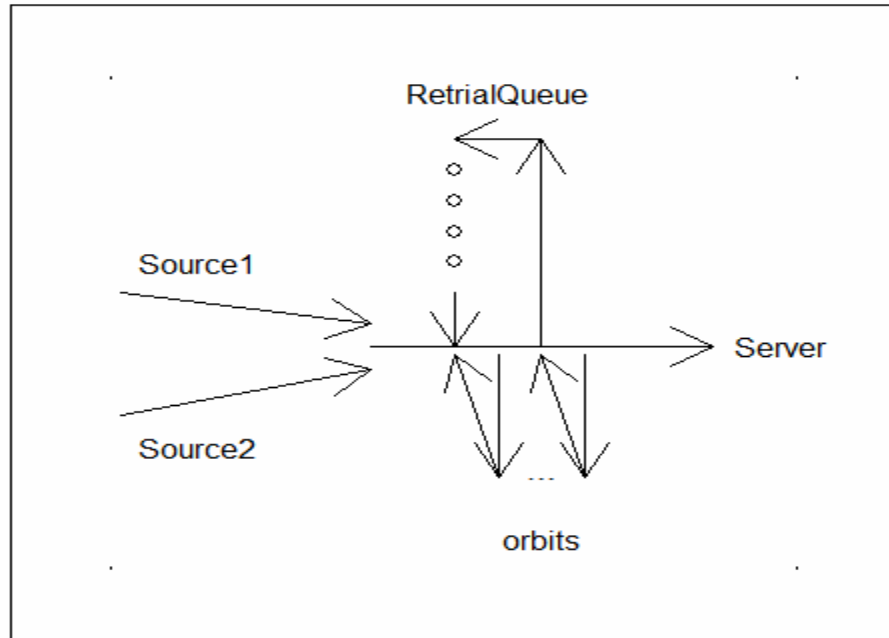
```
plot(-1:1,-1:1, type = "n", xlab = "Re", ylab = "Im")
K = 12; text(exp(1i * 2 * pi * c(2:1,12:3) / K), col = 2)
arrows(0,0,0,.9)
arrows(0,0,.7*cos(pi/6),.7*sin(pi/6))
```



EXAMPLE 20: Here I make a flow chart that I needed for a paper. Not too bad. The first four pairs of points (circles with radius zero) really just set up the size of the frame. There is likely a better way to do that but this worked.

```
symbols(c(0,0,8,8,4,4,4,4),c(-8,8,-  
8,8,2,3,4,5),circles=c(0,0,0,0,.1,.1,.1,.1),  
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)  
arrows(.1,1,3,0)  
arrows(.1,-3,3,-1.5)  
arrows(3,-.75,7,-.75)  
arrows(4,1,4,-.75)  
arrows(5,6,4,6)  
arrows(5,-.7,5,6)  
arrows(4.5,-5,4,-1)  
arrows(4.5,-1,4.5,-5)  
arrows(5.5,-5,5,-1)  
arrows(5.5,-1,5.5,-5)  
text(5,-7, expression(orbit))  
text(4.5,7.5,expression(RetrialQueue))  
text(7.8,-.7,expression(Server))  
text(1,2,expression(Source1))  
text(1,-4,expression(Source2))  
text(5,-5,expression(...))
```

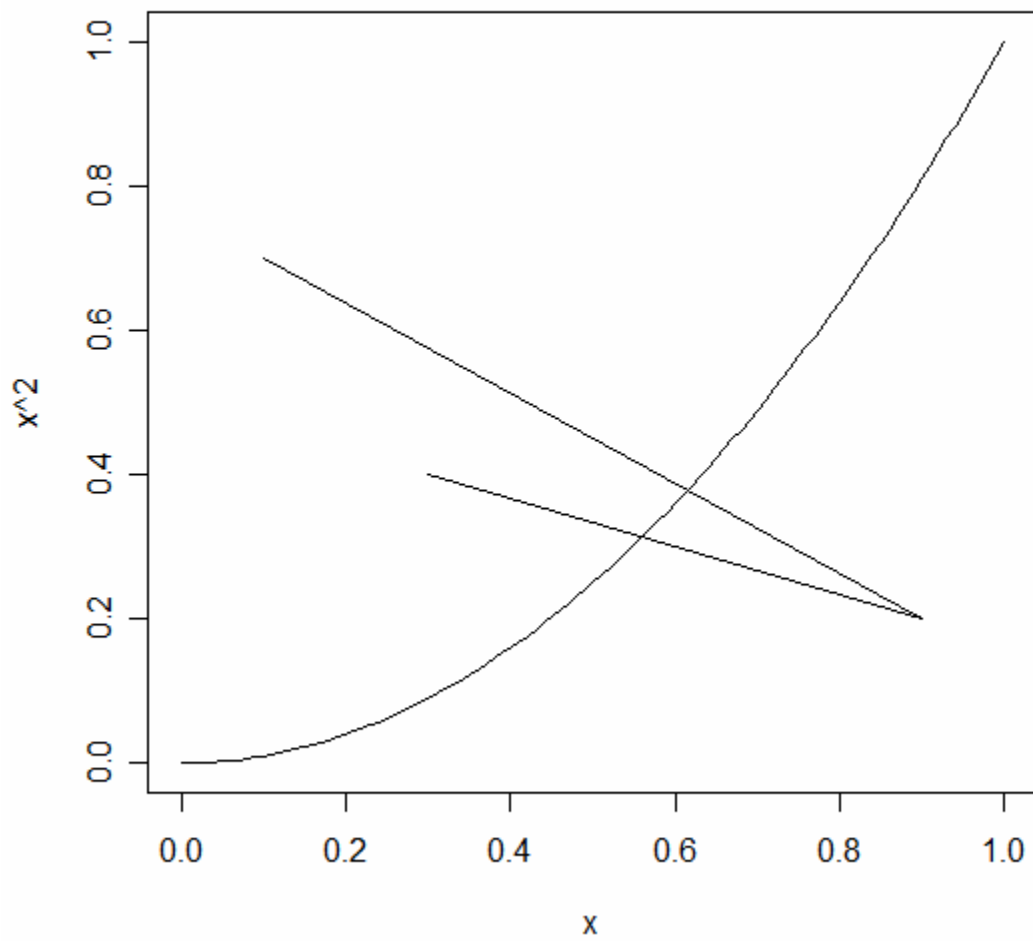
Output is



EXAMPLE 21. Here we add a line segments to an existing graph.

```
curve(x^2)
lines(c(.1,.9,.3),c(.7,.2,.4))
```

The line segments are from (.1, .7) to (.9, .2) to (.3, .4).



EXAMPLE 23: Writing graph commands as a function to allow easier editing.

```
graph1=function(n) {
symbols(c(1,7,2,3,4,5,6,4,5,6),c(1.2,3.5,2,2,2,2,2,2.8,2.8,2.8),
circles=c(.01,.01,.4,.4,.4,.4,.4,.4,.4),
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE);
arrows(2.1,2.2,2.8,2.2,length=.1) ;
arrows(4.1,2.2,4.8,2.2,length=.1) ;
arrows(5.1,2.2,5.8,2.2,length=.1)

arrows(3.5,2.6,2.2,2.3,length=.1);
arrows(4.5,2.6,3.3,2.3,length=.1) ;
arrows(5.5,2.6,4.2,2.3,length=.1);

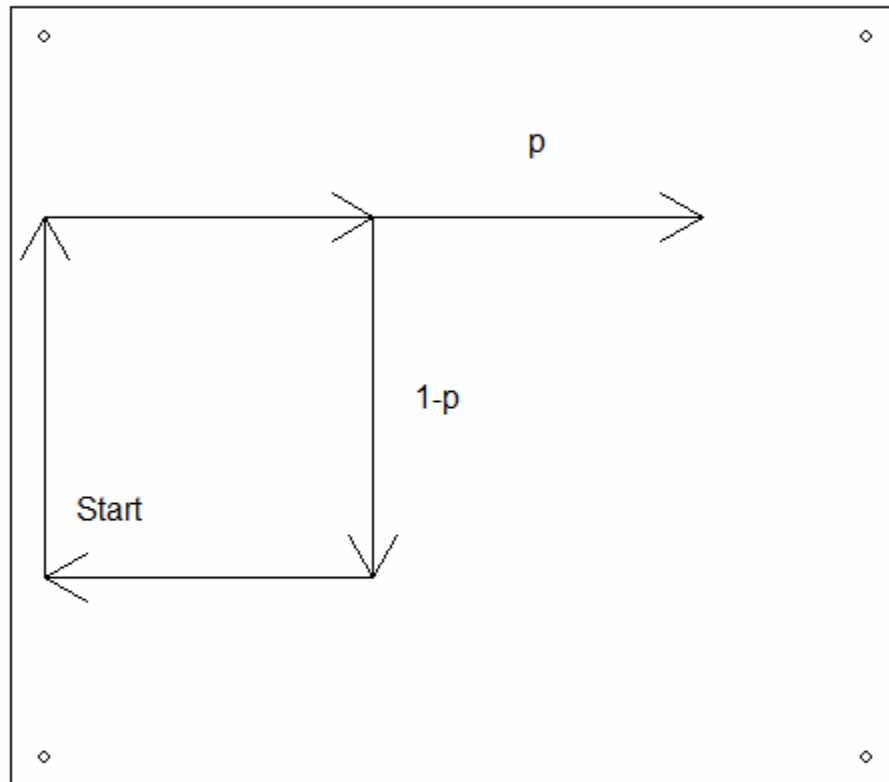
arrows(3.1,2.3,3.9,2.6,length=.1);

lines(c(5.8,4.8),c(1.8,1.5));arrows(4.8,1.5,4.1,1.8,length=.1)
lines(c(4.8,3.8),c(1.8,1.5));arrows(3.8,1.5,3.1,1.8,length=.1)
lines(c(3.8,2.8),c(1.8,1.5));arrows(2.8,1.5,2.1,1.8,length=.1)
text(2.5,2.3, expression(lambda));
text(4.5,2.3, expression(lambda)); text(5.5,2.3,
expression(lambda));
text(2.5,1.5, expression(mu)); text(3.5,1.5, expression(mu))
text(4.5,1.5, expression(mu));
text(2,2,0); text(3,2,1); text(4,2,2); text(5,2,3); text(6,2,4)
text(4,2.8,expression("2*")); text(5,2.8,expression("3*"));
text(6,2.8,expression("4*"))}
```

EXAMPLE 24:

```
x1=c(0 , 0, 2.5, 2.5)
y1=c(-.5,1.5,1.5, -.5)
plot(x1,y1 ,xaxt="n",yaxt="n",ann=FALSE)
arrows(1,1,2,1)
arrows(1,1,1,0)
arrows(0,0,0,1)
arrows(0,1,1,1)
arrows(1,0,0,0)
text(1.5,1.2,"p")
text(1.2,.5,"1-p")
text(.2,2,"Start")
```

Output is

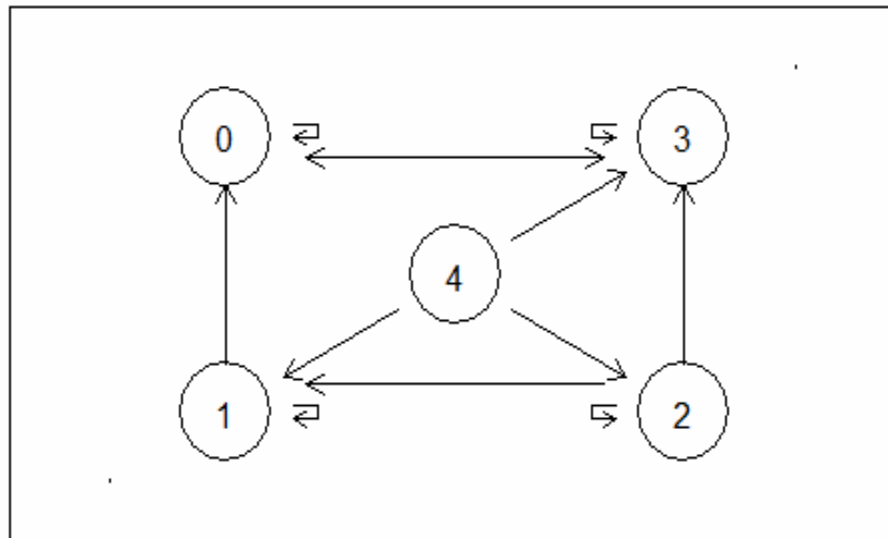


EXAMPLE 25: A Probability Transition Diagram

```

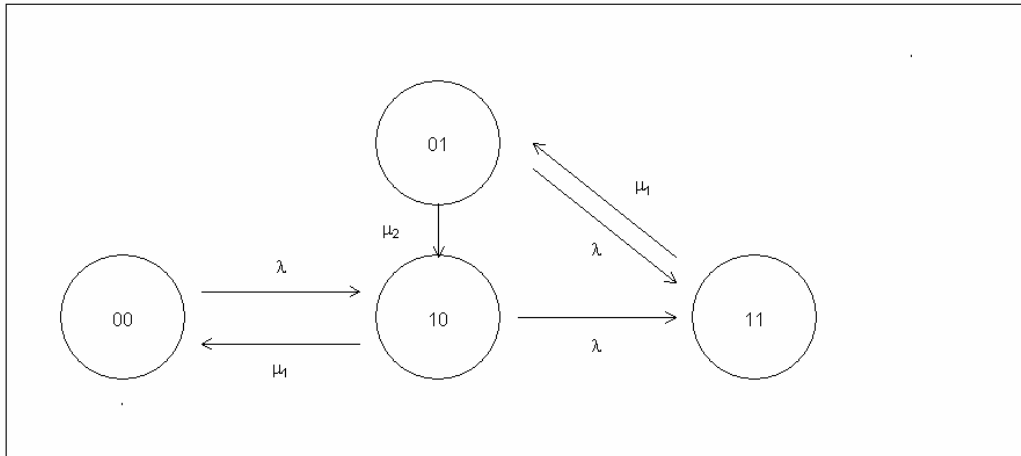
symbols(c(-1,5,0,0,2,4,4),c(-1,5,0,4,2,0,4),circles=c(.01,.01,.4,.4,.4,.4,.4),
 inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
arrows(.7,3.7,3.3,3.7, code=3, length=.1)
lines(c(.6,8,8),c(4.2,4.2,4.0), type="l")
arrows(.8,4.0,.6,4.0, length=.07)
lines(c(3.4,3.2,3.2),c(4.2,4.2,4.0), type="l")
arrows(3.2,4.0,3.4,4.0, length=.07)
arrows(0,.7,0,3.3, length=.1)
arrows(4,.7,4,3.3, length=.1)
arrows(3.3,4,.7,4, length=.1)
lines(c(.6,8,8),c(0.1,0.1,-.1), type="l")
arrows(.8,-.1,.6,-.1, length=.07)
lines(c(3.4,3.2,3.2),c(0.1,0.1,-.1), type="l")
arrows(3.2,-.1,3.4,-.1, length=.07)
arrows(2.5,2.5,3.5,3.5,length=.1);arrows(1.5,1.5,.5,.5,length=.1)
arrows(2.5,1.5,3.5,.5,length=.1)
text(0,0,"1");text(0,4,"0")
text(4,0,"2");text(4,4,"3")
text(2,2,"4")

```



EXAMPLE (Another state transition diagram)

```
symbols(c(1,3,3,5,1,6),c(1,1,3,1,0,4),circles=c(.4,.4,.4,.4,.01,.01),
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
arrows(1.5,1.3,2.5,1.3,length=.1); arrows(2.5,.7,1.5,.7,length=.1);
arrows(3.5,1,4.5,1,length=.1); arrows(3.0,2.3,3.0,1.7, length=.1);
arrows(3.6,2.7,4.5,1.4,length=.1);arrows(4.5,1.7,3.6,3.0,length=.1) ;
text(1,1,"00"); text(3,1,"10"); text(3,3,"01"); text(5,1,"11");
text(2,1.6,expression(lambda)) ; text(2,.4, expression(mu[1]))
text(2.7,2,expression(mu[2])) ; text(4,.7, expression(lambda))
text(4.3,2.5,expression(mu[1])) ; text(4,1.8, expression(lambda))
```

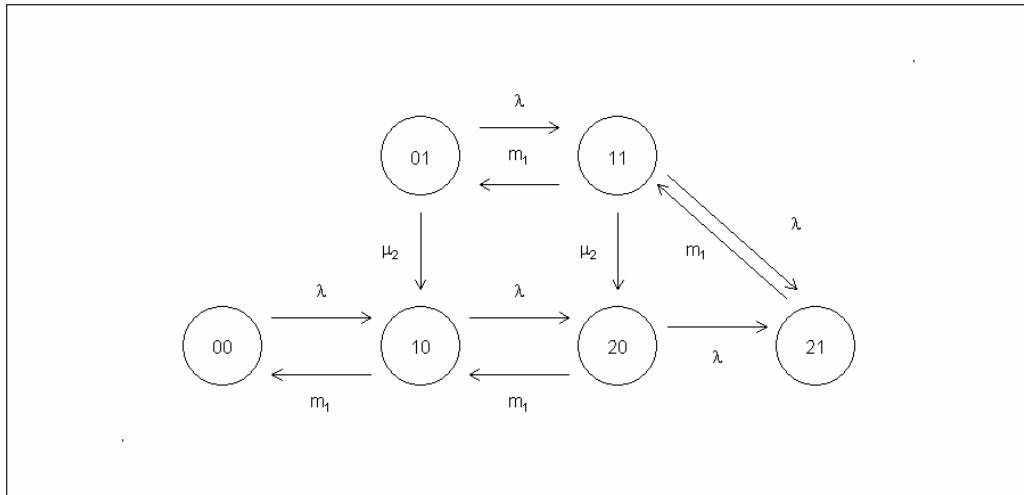


EXAMPLE (More transition diagrams)

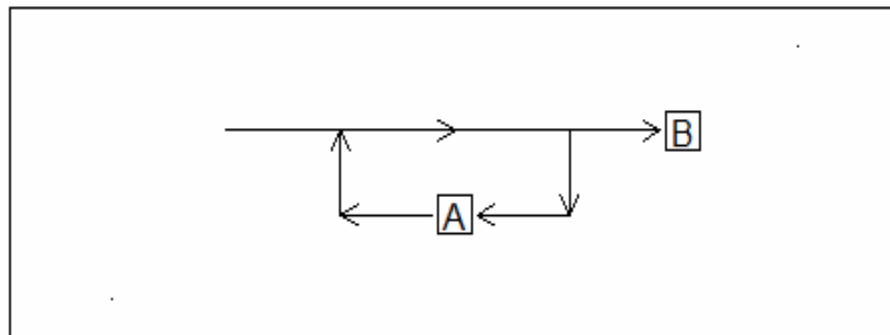
```

symbols(c(1,3,5,7,3,5,8,0),c(1,1,1,1,3,3,4,0),circles=c(.4,.4,.4,.4,.4,.01,.01),
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
arrows(1.5,1.3,2.5,1.3,length=.1); arrows(3.5,1.3,4.5,1.3,length=.1);
arrows(2.5,.7,1.5,.7,length=.1); arrows(4.5,.7,3.5,.7,length=.1);
arrows(5.5,1.2,6.5,1.2,length=.1);
arrows(3.6,3.3,4.4,3.3,length=.1); arrows(4.4,2.7,3.6, 2.7,length=.1);
arrows(3,2.4, 3, 1.6,length=.1); arrows(5, 2.4, 5,1.6,length=.1);
text(1,1,"00"); text(3,1,"10"); text(5,1,"20"); text(7,1,"21");
text(3,3,"01"); text(5,3,"11")
text(2,1.6,expression(lambda)) ; text(2,.4, expression(m[1]))
text(4,1.6,expression(lambda)) ; text(4,.4, expression(m[1]))
text(6,.9, expression(lambda))
text(4,3.6,expression(lambda)) ; text(4,3,expression(m[1])) ;
text(4,3.6,expression(lambda)) ; text(4,3,expression(m[1])) ;
text(2.7,2,expression(mu[2])) ; text(4.7,2, expression(mu[2]))
arrows(6.7,1.5,5.4,2.7,length=.1); arrows(5.5,2.8,6.8,1.6,length=.1);
text(5.8, 2.0,expression(m [1])) ; text(6.8,2.3,expression(lambda)) ;

```

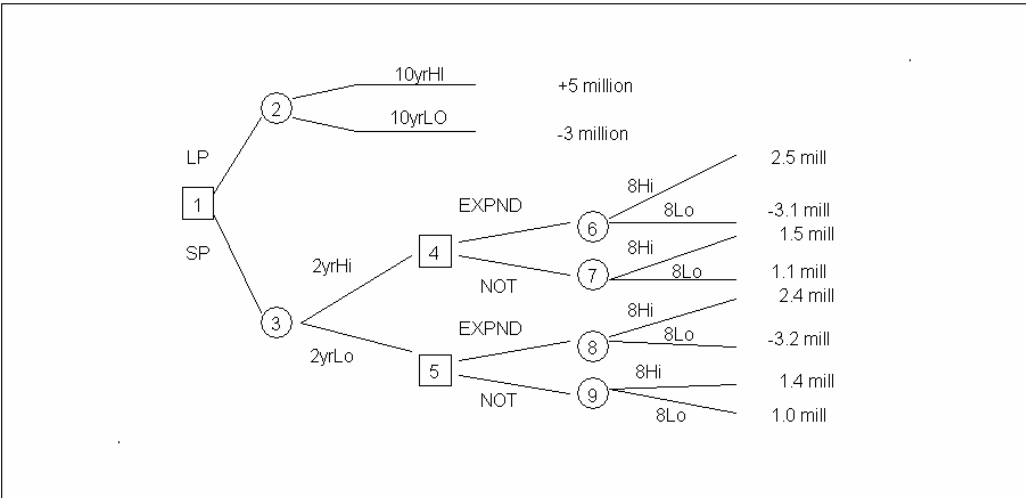


EXAMPLE: (Using squares)
symbols(c(0,3,5,6),c(0,1,2,3),squares=c(.001,.3,.3,.001),
inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
arrows(1,2,3,2,length=.1) ; arrows(3,2,4.8,2,length=.1)
arrows(4,2,4,1,length=.1) ; arrows(4,1,3.2,1,length=.1)
arrows(2.8,1,2,1,length=.1) ; arrows(2,1,2,2,length=.1)
text(3,1,"A"); text(5,2,"B")



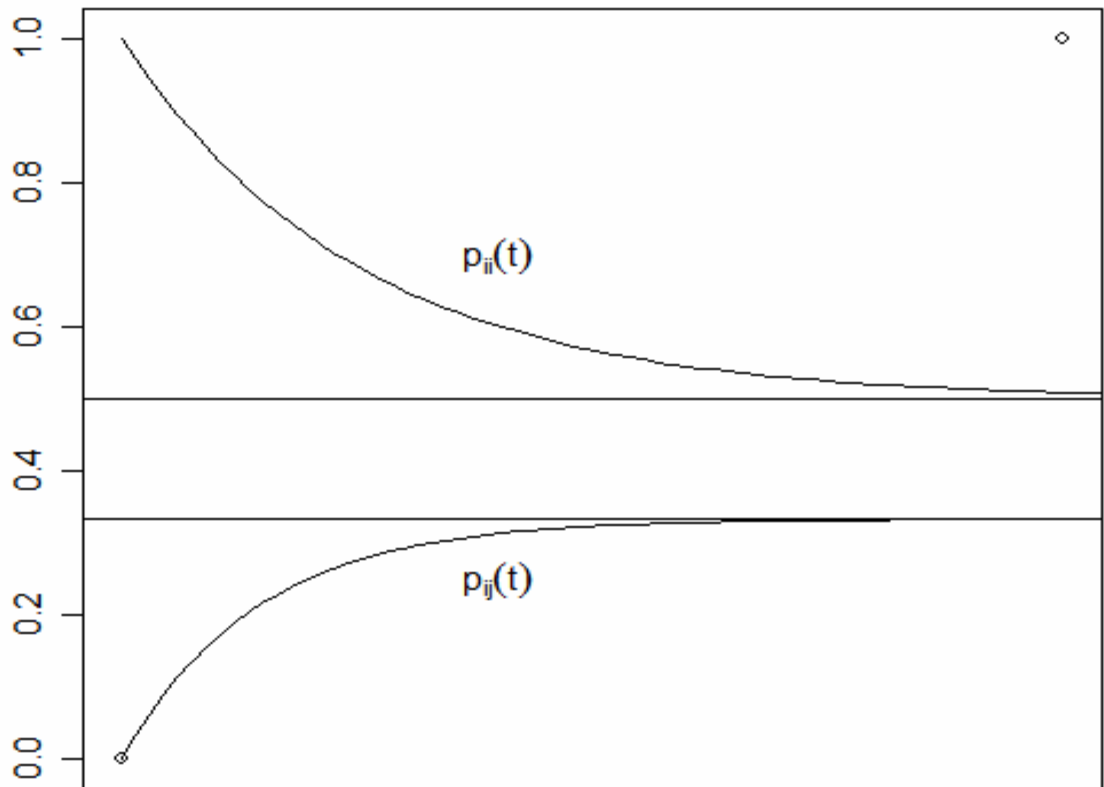
EXAMPLE (mixed symbols)

```
symbols( c( 0, 10, 1, 4, 4),
         c( 0, 8, 5, 4, 1.5 ),
         squares=c(.01, .01, .4, .4, .4 ),
         inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
symbols( c(2, 2,6,6,6,6), c( 7, 2.5, 4.5,3.5,2,1 ), circles=c(.2, .2,.2,.2,.2,.2),
         inches=FALSE,xaxt="n",yaxt="n",ann=FALSE,add=T)
lines(c(1.2,1.8),c(5.2,6.8)); lines(c(1.2,1.8),c(4.8,2.7))
text(1,5,1); text(2,7,2); text(2, 2.5,3); text(4,4,4); text(4,1.5,5)
text(6,4.5,6); text(6,3.5,7); text(6,2,8); text(6,1,9)
lines(c(2.2, 3),c(7.2,7.5));lines(c(3, 4.5),c(7.5,7.5))
lines(c(2.2,3),c(6.8,6.5));lines(c(3,4.5),c(6.5,6.5))
text(1,6,"LP"); text(1,4,"SP")
text(3.8,7.7,"10yrHI"); text(3.8,6.8,"10yrLO")
text(6,7.5," +5 million"); text(6,6.5,"-3 million")
lines(c(2.3,3.7),c(2.5,3.9)); lines(c(2.3,3.7),c(2.5,1.9))
text(2.7,3.7,"2yrHi"); text(2.7,1.8, "2yrLo")
lines(c(4.3,5.7), c(4.2,4.6)); lines(c(4.3,5.7), c(3.9,3.5))
lines(c(4.3,5.7), c(1.7,2.1)); lines(c(4.3,5.7), c(1.4,1.0))
text(4.7,5,"EXPND"); text(4.8, 3.3,"NOT")
text(4.7,2.4,"EXPND"); text(4.8, 0.9,"NOT")
lines(c(6.2,7.8),c(4.7,6)); lines(c(6.2,7.8),c(4.6,4.6))
lines(c(6.2,7.8),c(3.4,4.3)); lines(c(6.2,7.8),c(3.4,3.4))
lines(c(6.2,7.8),c(2.2,3.0)); lines(c(6.2,7.8),c(2.1,2.0))
lines(c(6.2,7.8),c(1.1,1.2));lines(c(6.2,7.8),c(1.1,.6))
text(6.6,5.4,"8Hi"); text(7.1,4.9,"8Lo")
text(6.6,4.1,"8Hi"); text(7.2,3.6,"8Lo")
text(6.6,2.8,"8Hi"); text(7.1,2.3,"8Lo")
text(6.7,1.5,"8Hi"); text(7.0,.6,"8Lo")
text(8.6,6, "2.5 mill"); text(8.6,4.9,"-3.1 mill")
text(8.7,4.4, "1.5 mill"); text(8.6,3.6,"1.1 mill")
text(8.7,3.1, "2.4 mill"); text(8.6,2.2,"-3.2 mill")
text(8.7,1.3, "1.4 mill"); text(8.6,0.6,"1.0 mill")
```



EXAMPLE:

```
plot(c(0, 10),c(0,1),ann=F,xaxt="n")  
curve((1-exp(-.7*(x)))/3, 0, 15, add=T,xaxt="n")  
curve(1-(1-exp(-.4*x))/2,0,15,add=T,ann=F)  
abline(.5,0);abline(.333,0)  
text(4,.7,expression(p[ii](t)))  
text(4,.25,expression(p[ij](t)))
```

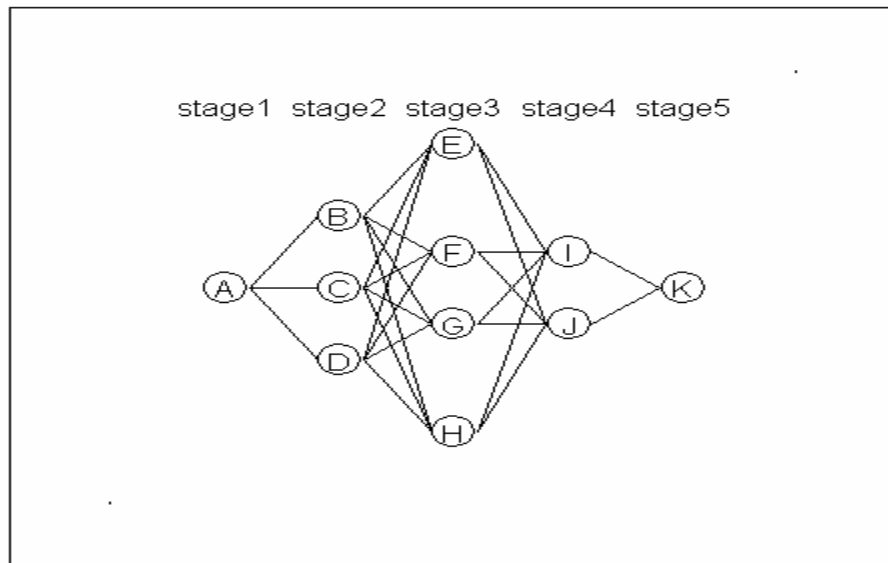


EXAMPLE:

```

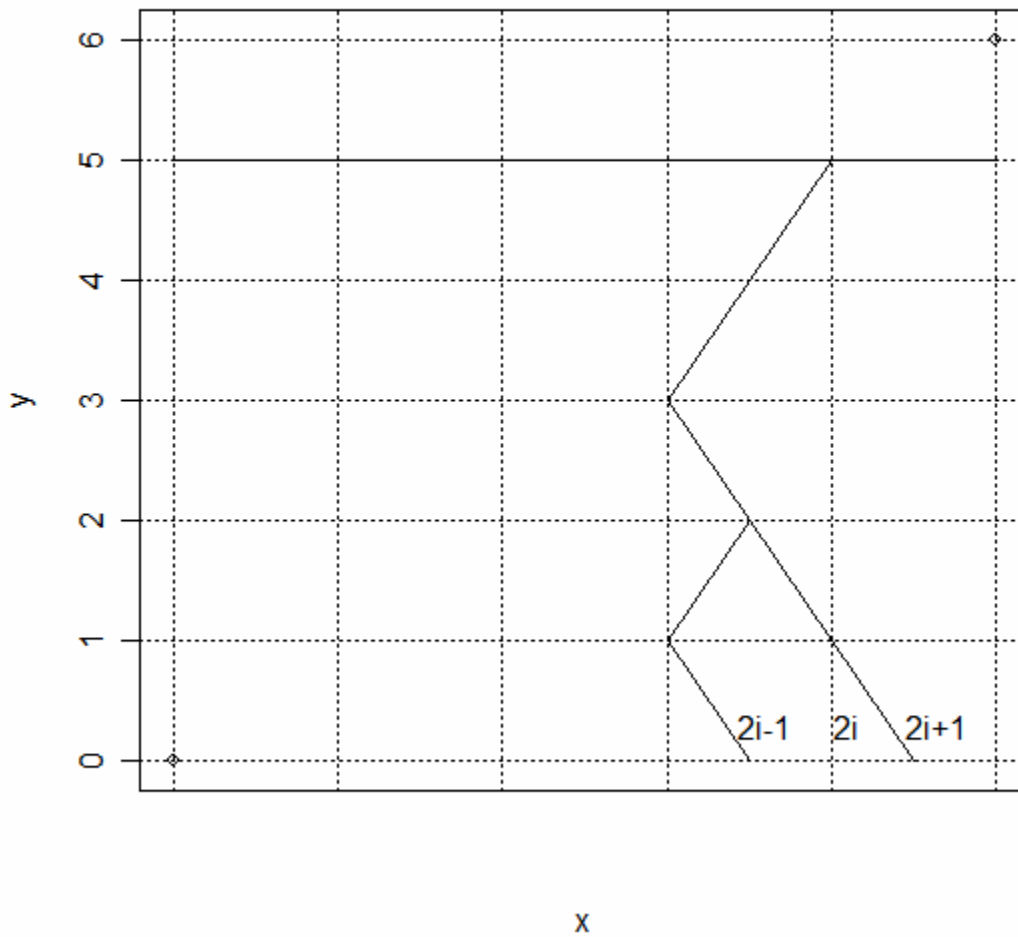
symbols( c(0, 6, 1, 2, 2,      2, 3, 3, 3, 3,      4,4,5),
         c( 0, 6, 3, 2, 3,      4, 1, 2.5, 3.5, 5,  2.5, 3.5, 3 ),
  circles= c(.01, .01, .2, .2, .2,  .2,.2,.2,.2,.2,  .2,.2,.2 ),
inches=FALSE, xaxt="n", yaxt="n", ann=FALSE)
text(1,3,"A"); text(2,2,"D"); text(2, 3,"C"); text(2,4,"B"); text(3,1,"H");
text(3,2.5,"G"); text(3,3.5,"F"); text(3,5,"E"); text(4,2.5,"J")
text(4,3.5,"I"); text(5,3,"K")
lines(c(1.2,1.8),c(3,2)); lines(c(1.2,1.8),c(3,3));lines(c(1.2,1.8),c(3,4));
lines(c(2.2,2.8),c(4,5));lines(c(2.2,2.8),c(4,3.5));lines(c(2.2,2.8),c(4,2.5))
lines(c(2.2,2.8),c(4,1));
lines(c(2.2,2.8),c(3,5));lines(c(2.2,2.8),c(3,3.5));lines(c(2.2,2.8),c(3,2.5))
lines(c(2.2,2.8),c(3,1));
lines(c(2.2,2.8),c(2,5));lines(c(2.2,2.8),c(2,3.5));lines(c(2.2,2.8),c(2,2.5))
lines(c(2.2,2.8),c(2,1));
lines(c(3.2,3.8),c(5,3.5));lines(c(3.2,3.8),c(3.5,3.5));lines(c(3.2,3.8),c(2.5,3.5));
lines(c(3.2,3.8),c(1,3.5));
lines(c(3.2,3.8),c(5,2.5));lines(c(3.2,3.8),c(3.5,2.5));lines(c(3.2,3.8),c(2.5,2.5));
lines(c(3.2,3.8),c(1,2.5));
lines(c(4.2,4.8),c(3.5,3));lines(c(4.2,4.8),c(2.5,3));
text(1,5.5,"stage1"); text(2,5.5,"stage2"); text(3,5.5,"stage3");
text(4,5.5,"stage4"); text(5,5.5,"stage5");

```



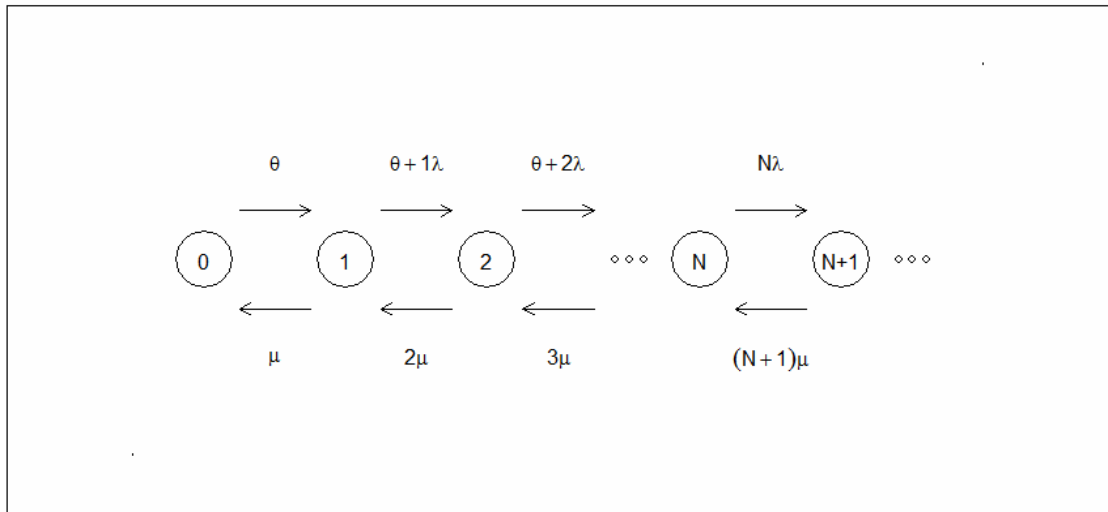
EXAMPLE: Note the use of grid and spacing or overlays.

```
x=c(0,10); y=c(0,6)
plot(x,y, xaxt="no")
lines(c(6,7),c(1,0))
lines(c(6,7),c(1,2))
lines(c(6,9),c(3,0))
lines(c(6,8),c(3,5))
grid(col="black")
lines(c(0,10),c(5,5))
text(9.3,3,"2i+1")
text(8.2,3,"2i")
text(7.2,3,"2i-1")
```



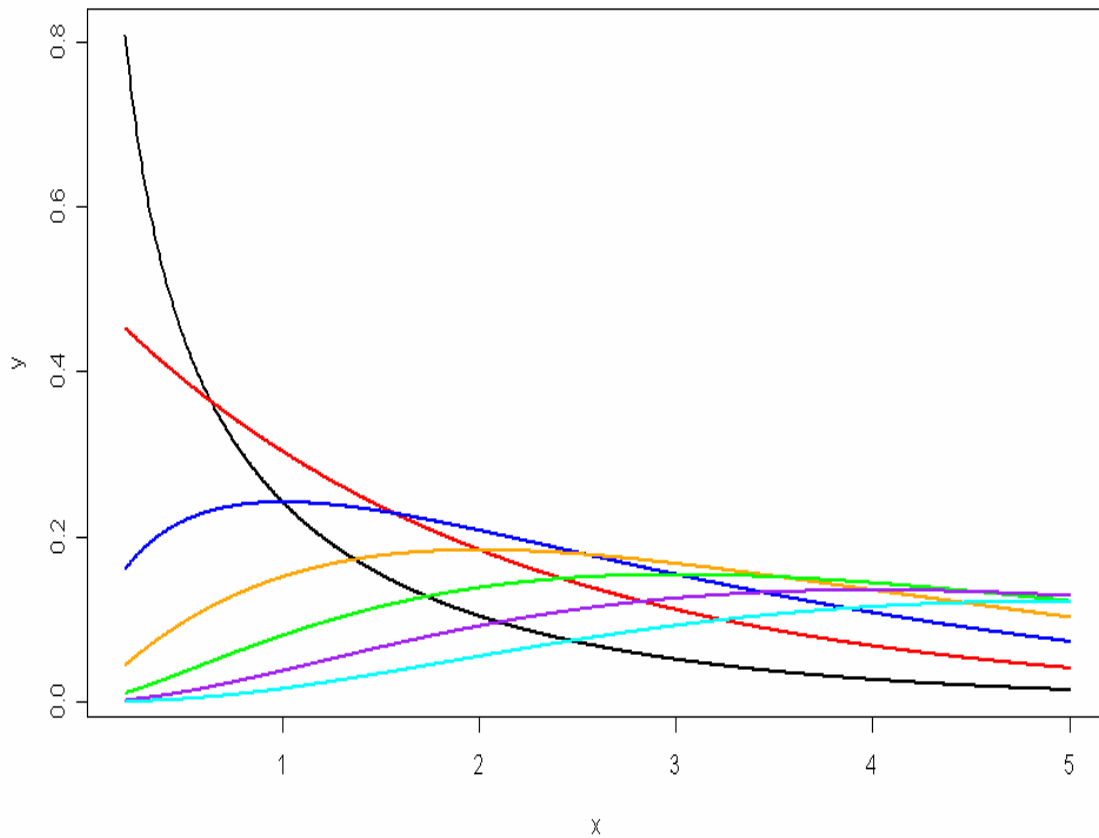
EXAMPLE: Another flow chart.

```
symbols(c(0,12,1,3,5,8,10,6.8,7,7.2,10.8,11,11.2),c(0,4,2,2,2,2,2,2,2,2,2,2,2,2,2,2),circles=c(.01,.01,.4,.4,.4,.4,.4,.4,.05,.05,.05,.05,.05,.05),inches=FALSE,xaxt="n",yaxt="n",ann=FALSE)
arrows(1.5,2.5,2.5,2.5,length=.1); arrows(3.5,2.5,4.5,2.5,length=.1);
arrows(5.5,2.5,6.5,2.5,length=.1); arrows(8.5,2.5,9.5,2.5,length=.1);
arrows(9.5,1.5,8.5,1.5,length=.1); arrows(6.5,1.5,5.5,1.5,length=.1);
arrows(4.5,1.5,3.5,1.5,length=.1); arrows(2.5,1.5,1.5,1.5,length=.1);
text(1,2,0); text(3,2,1); text(5,2,2); text(8,2,"N"); text(10,2,"N+1")
text(2,3,expression(theta)); text(4,3,expression(theta+1* lambda));
text(6,3,expression(theta+2*lambda)); text(9,3,expression(N* lambda));
text(2,1,expression(mu)); text(4,1,expression(2*mu));
text(6,1,expression(3*mu)); text(9,1,expression((N+1)*mu));
```



EXAMPLE: Colors and thick lines. We plot the chi-square pdf for various parameters.

```
x=(40:1000)/200
y1=dchisq(x,1)
plot(x,y1, ylab="y",type="l",lwd=2)
y2=dchisq(x,2)
lines(x,y2, col="red",lwd=2)
y3=dchisq(x,3)
lines(x,y3, col="blue",lwd=2)
y4=dchisq(x,4)
lines(x,y4, col="orange",lwd=2)
y5=dchisq(x,5)
lines(x,y5, col="green",lwd=2)
y6=dchisq(x,6)
lines(x,y6, col="purple",lwd=2)
y7=dchisq(x,7)
lines(x,y7,col="cyan",lwd=2)
label(3,4,expression(chi^2 pdf))
```



EXAMPLE: Draw a vector diagram

```
> plot(c(-1,5),c(-1,4),t="n",xaxt="n",yaxt="n",ann=FALSE)
  % This sets the frame size and removes all labels.
> abline(0,0)
> abline(0,1000000) % One way of getting a vertical axis
> arrows(0,0,1,3,length=.1)
> arrows(0,0,4,1,length=.1)
> lines(c(1,5),c(3,4),col="red")
> lines(c(4,5),c(1,4),col="red")
> arrows(0,0,5,4,length=.1, col="blue")
> text(1,2,"a")
> text(2,2,"a+b")
> text(3,1,"b")
```

