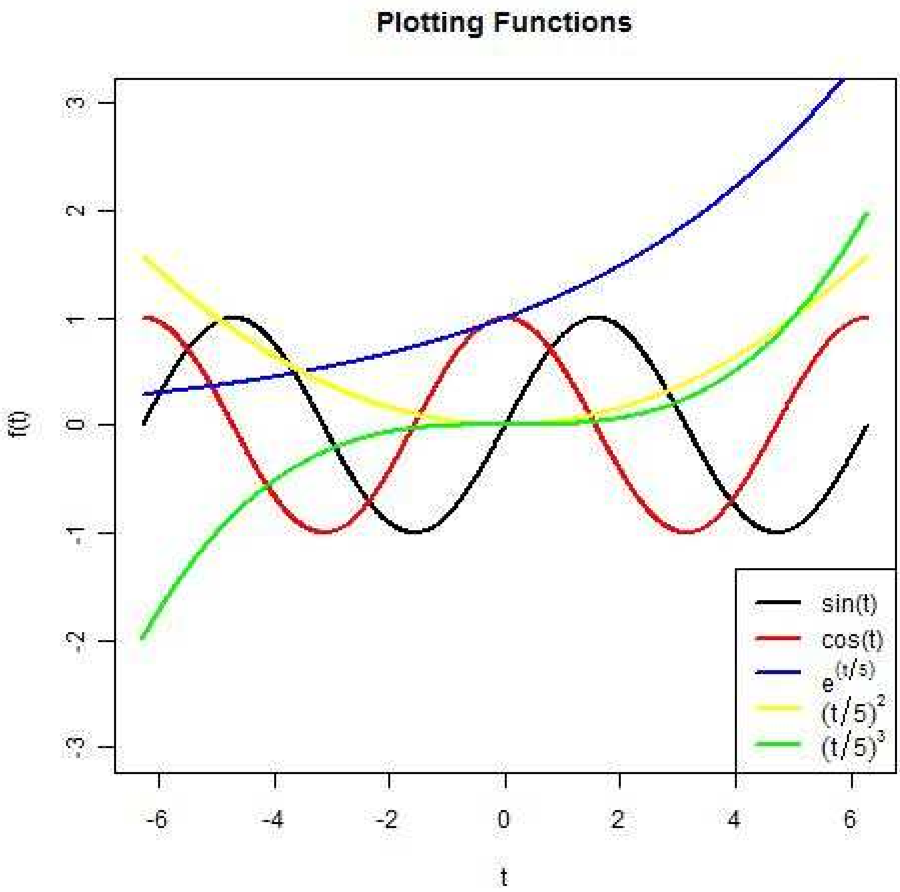
DYNAMICS II: Exercise 1

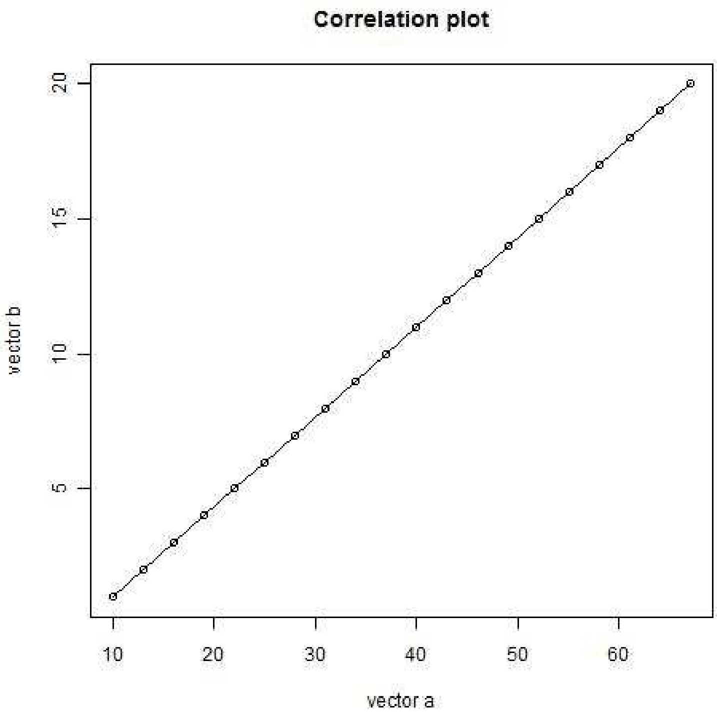
2.)

```
t <- seq(-2*pi, 2*pi, by=0.01)

jpeg("functions.jpg")

plot(t, sin(t), xlab="t", ylab="f(t)", ylim=c(-3,3), xlim=c(-2*pi,2*pi), main="Plotting Functions",
type="l", lwd=2)
lines(t, cos(t), col="red", lwd=2)
lines(t, exp(t/5), col="blue", lwd=2)
lines(t, (t/5)^2, col="yellow", lwd=2)
lines(t, (t/5)^3, col="green", lwd=2)
legtxt <- c("sin(t)", "cos(t)", expression(e^(t/5)), expression((t/5)^2), expression((t/5)^3))
cols <- c("black", "red", "blue", "yellow", "green")
widths <- c(2,2,2,2,2)
legend(x="bottomright", legend=legtxt, col=cols, lwd=widths)

dev.off()
```



3.)

```
b <- seq(20)
a = 3*b+7
cor(a,b)
jpeg("corrplot.jpg")
plot(a,b,type="o",main="Correlation plot",xlab="vector a",ylab="vector b")
dev.off()
```



COMMENTS:

Two vectors a and b have been set up, which are related linearly. This means that when plotting one vector versus the other a perfect line is obtained. (Pearson's) Correlation coefficient indicates the degree of the linear relationship between two variables. It can vary from 0 to 1, the last indicating a perfect linear relationship. Regarding the elements of each vector as different realizations of a variable, the relation between the two variables the vectors represent is perfectly linear (as seen in the figure above) and therefore the correlation coefficient is unity, and positive as the slope of the line is positive.

4.)

```
vcov <- numeric()
for (i in 1:25)
{
        c <- rnorm(20)
        d <- rnorm(20)
        vcov[i] <- cor(c,d)
}

sample <- c(10,50,100,1000)
wcov <- vector()
for (j in sample)
{
        e <- rnorm(j)
        f <- rnorm(j)
        wcov <- c(wcov, cor(e, f))
}
vcov
wcov
```

```
> vcov
 [1] 0.22847113  -0.29687340  -0.14328974  -0.36319075  0.29041457  0.08699686
 [7] 0.22089522  0.18722304  0.14267178  -0.24612612  -0.27291385  -0.03009211
[13] 0.09077990  -0.03126880  0.13218594  0.16551527  0.09657156  -0.24652687
[19] -0.04011931  0.18398576  -0.02560334  -0.01589366  0.42442738  -0.20964352
[25] -0.26573349

> wcov
[1] 0.29892805  0.04903600  0.02441405  0.08149827
```

COMMENTS:

In first step, two vectors of length 20 composed of random numbers have been set up in order to afterwards compute the correlation between them. This operation has been repeated 25 times and the correlation coefficients obtained have been loaded in a new vector called “vcov”. Random number sequences are supposed not to follow any pattern, so correlation coefficients close to 0 are expected. However, for small number of points, some degree of linearity might appear (the extreme case would be for 2 data points, which always yield a line), as seen in the values obtained in “vcov”; some of them are quite far from the expected 0.

The length of the vectors has been varied, and once again the correlation coefficient has been computed and loaded in a new vector “wcov”. In general, the longer the vectors’ length the closer the correlation coefficient is to 0. The result shown above, however, differs from the expectation as the last correlation coefficient in “wcov” corresponding to the longest sample length is greater than the previous one. This might be due to the particular random number sequence generated; as seen in the previous step, results vary from one generation to other. I wonder whether this has something to be with the fact that random numbers generated trough an algorithm are not truly random numbers and are usually generated from a “seed”.

5.)

```
sample <- c(10,50,100,1000)
m <- numeric()
for (j in sample)
{
        cor.val <- vector()
        for (i in 1:100)
        {
                c <- rnorm(j)
                d <- rnorm(j)
                cor.val[i] <- cor(c,d)
        }

        m <- c(m,mean(cor.val))
        x<-rep(m[length(m)],times=100)
        y<-seq(0,99)
        name <- paste(paste("hist",j),".png")
        jpeg(name)
        hist(cor.val, xlab="Correlation Coefficient", col="lightblue")
        lines(x,y,col="red",lwd=2)
        legend(x="topright",legend="mean value",col="red",lwd=2)
        dev.off()
}
m
```

```
> m
[1] -0.0772334738   -0.0134492073   0.0054024835   0.0005939317
```

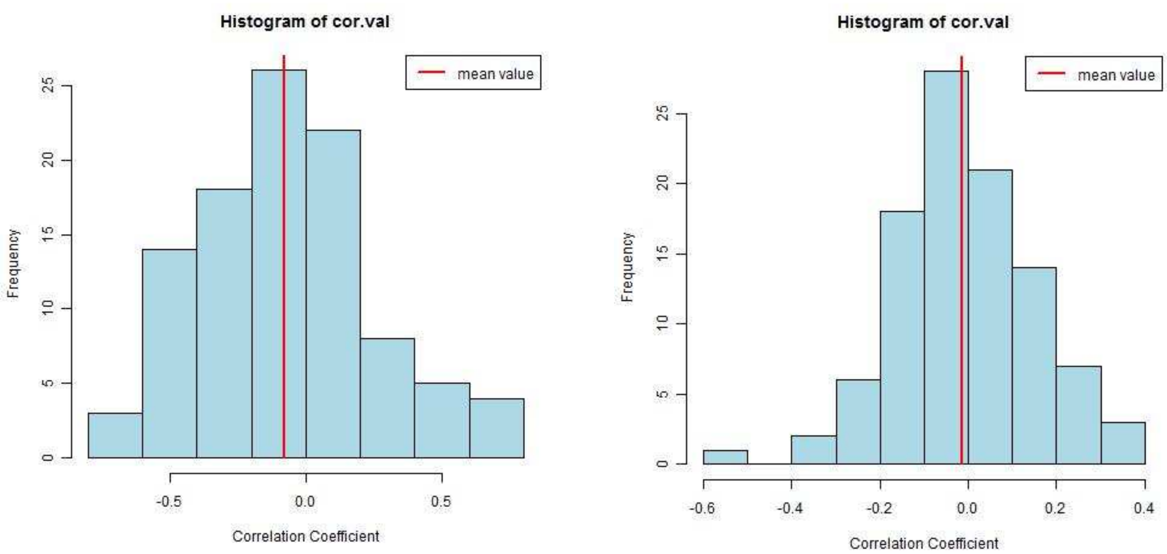


Fig. 1: Histograms of the correlation coefficient between two vectors of random numbers. Length of the vectors: 10 (left), 50 (right)

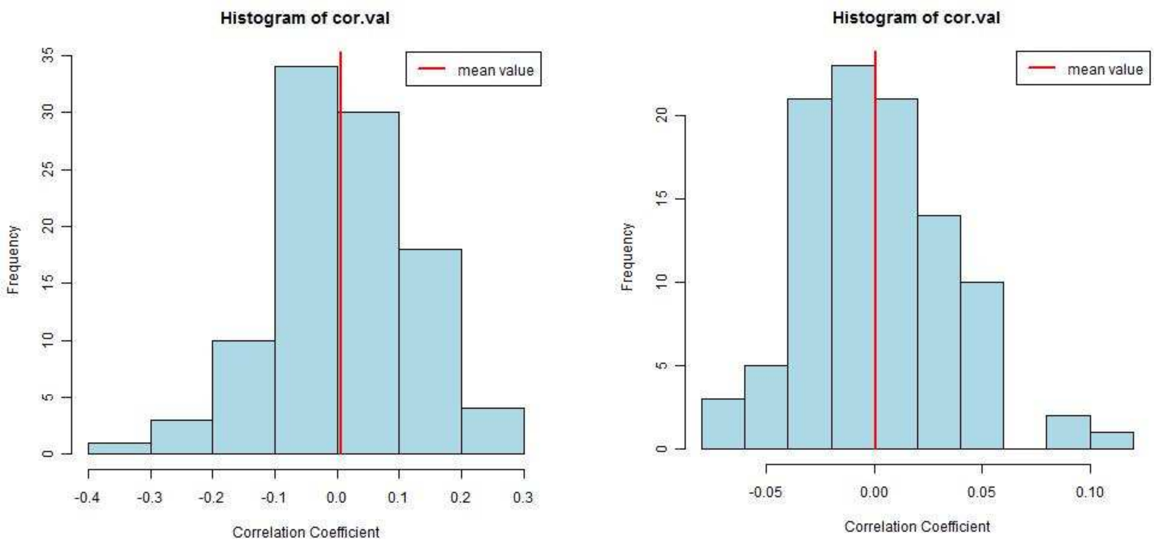


Fig. 2: Histograms of the correlation coefficient between two vectors of random numbers. Length of the vectors: 100 (left), 1000 (right)

COMMENTS:

As said in the previous exercise, the correlation coefficient varies from one generation of random numbers to another.

In the present exercise, 100 different generations have been done for each vector length, the correlation coefficient has been calculated in each generation and loaded in a new vector.

Four vectors of coefficients have been computed, one per each length: 10, 50, 100, 1000. For each of these vectors the mean has been computed and a histogram has been plotted, as seen in the figures above. The bars in the histograms represent the frequency of occurrence of the correlation coefficient lying inside a particular value interval.

Now, it is possible to check how the longer the vectors' length the closer the correlation coefficient is to 0, which was not possible with the single values obtained in the previous exercise.

Looking at the histograms, it can be seen how when increasing the vectors' length the scale changes and most of the values lie closer to 0, together with the mean value, represented by a red line.

6.)

```
cor.val <- vector()
r <- -3
for (i in 1:100)
{
	c <- rnorm(100)
	d <- r*c+rnorm(100)
	cor.val[i] <- cor(c,d)
}

m <- mean(cor.val)
x<-rep(m,times=100)
y<-seq(0,99)
jpeg("newhist.jpg")
hist(cor.val, xlab="Correlation Coefficient", col="lightblue")
lines(x,y,col="red",lwd=2)
legend(x="topright",legend="mean value",col="red",lwd=2)
dev.off()

m
```

```
> m
[1] -0.9510608
```

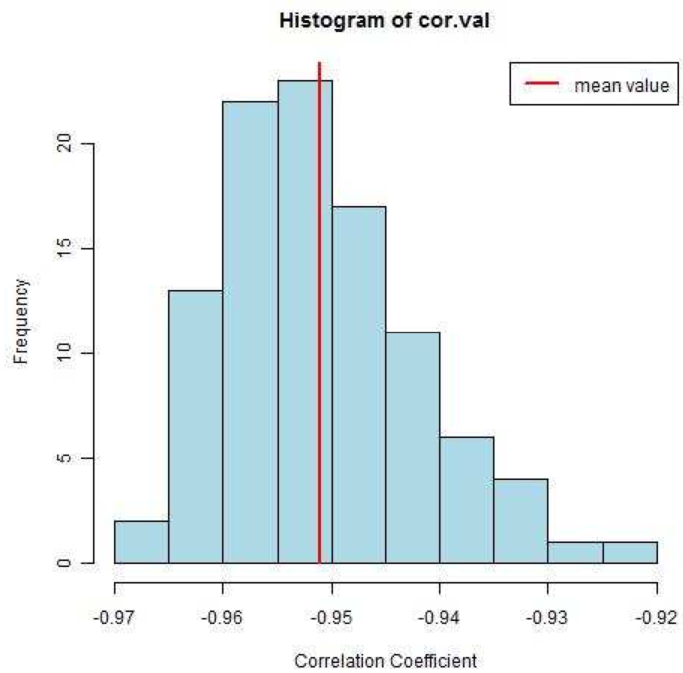
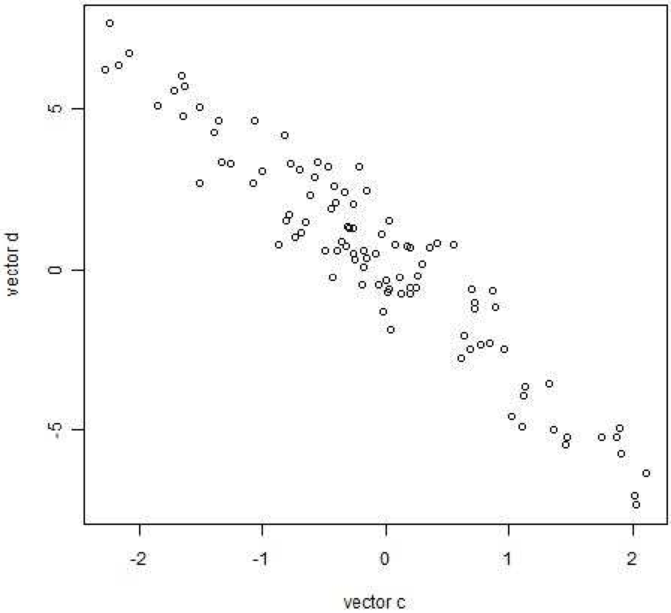


Fig. 3: *Histogram of the correlation coefficient between two partly linear dependent vectors. Length of the vectors: 100*

COMMENTS:

Two partly linear dependent vectors of length 100 have been set up. The first of them (vector c) has been randomly generated and the second one (vector d) has been built up from a linear relationship to the first one but adding a set of random values (d<-r*c+rnorm(100)). The value of the slope of the line r has been chosen to be -3. These two vectors could represent a set of realizations (measurements) of a random variable each, where the two variables are linearly correlated but their realizations show deviations from the straight line due to measurement errors assumed as random with standard deviation of 1 from the mean 0. This can be seen if the vectors are plotted versus each other:



The procedure has been repeated 100 times and for each pair of vectors the correlation coefficient has been computed. The mean value of these coefficients has turned out to be around -0.95; close to unity due to the linear relationship, but not exactly 1 due to the effect of the random deviations and negative because of the choice of r. As seen in the histogram most values lie between -0.94 and -0.96.

10/04/2011
Patricia Altube Vázquez