

# Referències ràpides per a l'R

## Taules de sintaxis del llenguatge R

### Elementals

Comanda	Comentari sobre la comanda	Exemple( <i>Input</i> )	( <i>ouput</i> )
#	Serveix per ficar comentaris.	#This is a code comment	
*	Operador producte.	3*2	[1] 6
sqrt()	Funció arrel quadrada.	sqrt(196)	[1] 14
pi	Quocient de la longitud d'una circumferència entre el doble del seu radi.	-pi	[1] -3.141593
e	Com prefixe potencia de deu	3e2 #Thermopylae number	[1] 300
as.complex(a+bi)	bi amb a i b reals	as.complex(-4) #-4 vist com a nombre complex	[1] -4+0i
i	Unitat Complexa	3i	[1] 0+3i
""	Per donar una cadena.	"En el Nom de Edmon"	[1] "En el Nom de Edmon"
T	Variable Booleana. Veritat.	T #TRUE	[1] TRUE
F	Variable Booleana. Fals.	F #FALSE	[1] FALSE
NA	Not available, hi ha alguna dada que falta o que no té sentit en algun lloc.	NA #NADENA	[1] NA
+, -, *, /, ^	Operadors bàsics, en igual ordre, suma, resta, producte, quocient, potència. ( <b>Nota:</b> El vectors o llistes és sumen component a component).	(1+2-3*4/5^6)	[1] 2.999232

### Successions

Comanda	Comentari sobre la comanda	Exemple( <i>Input</i> )	( <i>ouput</i> )
rep()	<i>Repeat</i> , serveix per repetir el primer argument <i>n</i> vegades el segon argument	rep(":"), 2) #El doble de content	[1] ":" " ":"
1:10	Successions (llista) d'enters	1:10 #Comptar nombre de suspesos	[1] 1 2 3 4 5 6 7 8 9 10
5:1	Dóna els valors dels enters de gran a petit.	5:1	[1] 5 4 3 2 1

Comanda	Comentari sobre la comanda	Exemple( <i>Input</i> )	( <i>ouput</i> )
<code>seq(„by=)</code>	Serveix per fer llistes de nombres més elaborades. Podem pensar que ens dóna fins l'enèsim nombre d'una successió. De <code>-pi</code> fins a <code>pi</code> de 0.5 en 0.5.	<code>seq(-pi,pi,by=.5)</code>	[1] -3.1415927 -2.6415927 -2.1415927 -1.6415927 -1.1415927 -0.6415927 [7] -0.1415927 0.3584073 0.8584073 1.3584073 1.8584073 2.3584073 [13] 2.8584073
<code>seq(„length=)</code>	També podem ficar l'especificador <code>length=</code> per demanar 10 valors equiespaiats entre dos nombres de la recta real.	<code>seq(-pi,pi,length=10),</code>	<code>seq(-pi,pi,length=10)</code>
<code>seq(„by=, length=)</code>	Podem combinar les dues opcions per a fer una llista de <code>length</code> valors de <code>by</code> en <code>by</code> a partir d'un valor donat, en aquest cas 1.	<code>seq(1,by=.05,length=10)</code>	[1] 1.00 1.05 1.10 1.15 1.20 1.25 1.30 1.35 1.40 1.45
<code>espera=c("N","O")</code>	Podem definir un <code>array</code> /llista/vector de <i>Strings</i> .	<code>c("N",espera[2])</code>	[1] "N" "O"

## Arrays

Recordem la **Regla de reutilització**: si tenim dos vector de diferents mides `a` i `b`. Suposem que el vector més curt és `b`. Aleshores en fer l'operació `a*b`, `a+b`, `a-b`, etc, tenim que les operacions és fan *component a component* i quan s'acaben les *components* de `b` aleshores és reciclen en el mateix ordre fins acabar amb totes les components de `a`.

Exemple:

```
> a=c(1,2,3,4);b=c(-1,1);
> a*b
[1] -1 2 -3 4
```

Comanda	Comentari sobre la comanda	Exemple( <i>Input</i> )	( <i>ouput</i> )
<code>a &lt;- c(1,0,-1,1)</code>	Assignar valor vectorial <code>c(1,0,-1,1)</code> a la variable <code>a</code> . La <code>c</code> ve de concatenar.	<code>a&lt;- c(1,0,-1,1) #Forma de fer ho vella (previus versions R)</code>	
<code>a=c(1,0,-1,1)</code>	Assignar valor vectorial <code>c(1,0,-1,1)</code> a la variable <code>a</code> . Sintaxis més nova. Pot no funcionar en <i>obsolets</i> .	<code>a=c(1,0,-1,1)</code>	
<code>c(a,b)</code>	També serveix per concatenar vectors. Siguin <code>a=c(1,2)</code> i <code>b=c(3,4)</code> , llavors <code>c(a,b)</code> retornara el vector <code>c(1,2,3,4)</code> .	<code>c(c(1,2),c(3,4))</code>	[1] 1 2 3 4

Comanda	Comentari sobre la comanda	Exemple( <i>Input</i> )	( <i>ouput</i> )
a[2]	Retorna la segona component del vector <b>a</b> . El vector no està indexat per 0.	c(1,2)[1] #Suposem c(1,2)=a definit prèviament	[1] 1
c(1,2,4,4)[-3]	Retorna el vector c(1,2,4,4) menys la darrera component, és a dir, imprimeix [1] 1 2 4. El mateix podríem pensar per c(1,2,4,4)[-1] que imprimeix [1] 2 4 4.	c(1,2,4,4)[-3]	[1] 1 2 4
c(0,2,4,6,8,10,12,14)[3:4]	Retorna els components (o components) del vector desitjat.	c(0,2,4,6,8,10,12,14)[3:4] # Imprimira [1] 4 6	[1] 4 6
c(1,2,3,4,5)-1	Vector menys un nombre, és interpretar com restar/sumar la quantitat a cada una de les components.	c(1,2,3,4,5)-1	[1] 0 1 2 3 4
c(1,2,3,4,5)*2	Producte usual de <i>vector</i> per <i>escalar</i> .	c(1,2,3,4,5)*2	[1] 2 4 6 8 10

## Funcions Matemàtiques

Comanda	Comentari sobre la comanda	Exemple( <i>Input</i> )	( <i>ouput</i> )
max(a) i min(a)	Dóna el valor màxim i mínim de totes les components del vector <b>a</b> respectivament. Això també és pot veure com la <i>norma infinit</i>	min(c(8,3,7)); max(c(8,3,7));	[1] 3 [1] 8
sum(a)	Suma totes les components del vector <b>a</b> .	sum(c(8,3,7));	[1] 18
prod(a)	Dóna el producte de totes les components del vector <b>a</b> .	prod(c(8,3,7));	[1] 168
prod(a)	Dóna el producte de totes les components del vector <b>a</b> .	prod(c(8,3,7));	[1] 168
choose (n,k)	<i>n</i> sobre <i>k</i> , nombre combinatori, combinacions de agafar <b>k</b> elements sense ordre d'un conjunt de <b>n</b> elements.	choose(6,4);	[1] 15
factorial(n)	Dóna <i>n factorial</i> , $n!$ .	factorial(4); factorial(5); factorial(6);	[1] 24 [1] 120 [1] 720
lfactorial(n)	Serveix per obtenir directament el logaritme del valor factorial.	> lfactorial(4); lfactorial(5); lfactorial(6);	[1] 3.178054 [1] 4.787492 [1] 6.579251
g=function amb variables}	Certeix <i>per de</i> funcions amb funcions a partir de les anteriors.	f= function(x){x^2-1};f(2); a=c(1,2,3); f(a); # Aplica funció component a component	[1] 3 [1] 0 3 8

## Gràfics

Comentari sobre la comanda	Comanda	Gràfic
Representa els punts al pla que defineixen dos vector amb la mateixa longitud un per els valor de les abscisses <b>OX</b> i l'altre per les ordenades <b>OY</b> .	<code>x=c(3,1,5,3); y=c(1,3,3,5); plot(x,y);</code>	<code>#Veure</code> PlotTypeL.png
Representa una línia que uneix els punts al pla (per orde de component) que defineixen dos vector amb la mateixa longitud un per els valor de les abscisses <b>OX</b> i l'altre per les ordenades <b>OY</b> .	<code>x=c(3,1,5,3); y=c(1,3,3,5); plot(x,y, type="l");</code>	<code>#Veure</code> PlotTypeL.png
Per dibuixar fragments de funcions o corbes planes parametritzades. Observem que <code>col=2</code> fa referència a que el color per efectuar el gràfic és el <i>color 2</i> que tal com podem veure fa referència al color vermell.	<code>curve(exp(-x^2),from=-3,to=3,col=2);</code>	<code>#Veure</code> CurveBellGaus.png
Per introduir llegenda als gràfics de fragments de funcions o corbes planes parametritzades.	<code>curve(exp(-x^2),from=-3,to=3,col=2); legend('topright','exp(-x^2)',lty="n",col=2);</code>	<code>#Veure</code> CurveBellGaus.png
Per afegir la recta horitzontal i vertical que passa pel zero respectivament.	<code>curve(exp(-x^2),from=-3,to=3,col=2); abline(h=0); abline(v=0);</code>	<code>#Veure</code> CurveLines*
Podem prescindir de <code>for=i to= i</code> simplement ficar els dos valors per el paràmetre després de la funció <code>sin(x)</code> . <code>main=dibuix</code> permet ficar un títol al gràfic	<code>curve(sin(x),-pi,pi, main="dibuix");</code>	<code>#Veure</code> CosSin01.png
El <i>argument</i> <code>add=T</code> (T de True i F de False) permet ficar diversos gràfics junts. <code>lty=2</code> fa referència a <i>line type</i> és ha dir línia tipus 2, que tal com podem veure és una línia discontinua. <code>col=2</code> ens indica que el gràfic és dibuixarà amb color vermell.	<code>...; curve(cos(x),add=T,lty=2,col=2);</code>	<code>#</code> <code>#Veure</code> CosSin02.png
Com abans a <code>legend(...)</code> , <code>'topright'</code> indica la posició. El vector de <code>__Strings__</code> indicarà el nom de les dos gràfiques sobreposades. <code>lty=c(1,2)</code> indica que el primer element de la llegenda és amb línia contínua <code>lty=1</code> i el segon amb línia discontinua <code>lty=2</code> . El color és <code>col=1</code> per negre i <code>col=2</code> per al vermell, primera i segona gràfica respectivament.	<code>...; legend('topright',c('sin(x)','cos(x)'),lty=c(1,2),col=c(1,2));</code>	<code>#Veure</code> CosSin03.png
Amb <code>abline(h=0,v=0)</code> ; afegim les rectes horitzontals <code>h</code> i verticals <code>v</code> que passen per zero <code>=0</code> al gràfic. També podem pensar que això és el mateix que ficar els eixos <b>OX</b> i <b>OY</b> .	<code>...; abline(h=0,v=0);</code>	<code>#Veure</code> CosSin04.png

Comentari sobre la comanda	Comanda	Gràfic
Amb <code>point(...)</code> podem dibuixar punt. <code>-1:1</code> fa referencia a la regió del eix d'abscisses que agafem per dibuixar punt, amb punts equidistants. El vector <code>c(0,1,0.5)</code> indica el nombre de punts i els valors de les ordenades. <code>col=c(3,4,5)</code> indica els color ; <code>col=3</code> verd, <code>col=4</code> blau, <code>col=5</code> cian. <code>pch=</code> fa referencia a la forma dels punts. Potser passa el mateix que amb els colors <code>col=23</code> és el mateix que <code>col=7</code> , cicle <code>23-8*2=7</code> .	<code>...; points(-1:1,c(0,1,0.5),pch=c(18,20,22),col=c(3,4,5));</code>	#Veure CosSin05.png
Podem afegir més d'una gràfica en una finestra amb <code>par(mfrow=c(NumFiles,NumColum))</code> , en aquest cas dues files i una columna <code>par(mfrow=c(2,1))</code> .	<code>par(mfrow=c(2,1);</code> <code>curve(sin(x),-pi,pi,lty=3,col=3);</code> <code>curve(cos(x),lty=2,col=2);</code>	#Veure TwoPicturesInOne01 TwoPicturesInOne02
Podem definir el punt equiespaiats de 0 a $2\pi$ amb distància 0.1 al eix OX. Amb cercles o bé amb línies <code>type="l"</code> obtenint el mateix resultat (o bé augmentant la precisió establerta per defecte). També podriem fer servir la funció <code>lines(...)</code>	<code>x=seq(0,2*pi,by=0.1);</code> <code>plot(x,sin(x)); plot(x,sin(x),</code> <code>type="l");</code>	#Veure PlotWithSeq.png

```
x=c(3,1,5,3); y=c(1,3,3,5); plot(x,y);
```

```
x=c(3,1,5,3); y=c(1,3,3,5); plot(x,y, type="l");
```

```
curve(exp(-x^2),from=-3,to=3,col=2)
```

```
curve(exp(-x^2),from=-3,to=3,col=2); legend('topright','exp(-x^2)',lty=1,col=2);
```

```
curve(sin(x),-pi,pi, main="dibuix"); curve(cos(x),add=T,lty=2,col=2) ;legend('topright',c('sin(x)','cos'),
abline(h=0,v=0); points(-1:1,c(0,1,0.5),pch=c(18,20,22), col=c(3,4,5));
```

```
curve(sin(x),-pi,pi, main="dibuix");
```

```
curve(cos(x),add=T,lty=2,col=2);
```

```
legend('topright',c('sin(x)','cos(x)'),lty=c(1,2),col=c(1,2));
```

```
abline(h=0,v=0);
```

```
points(-1:1,c(0,1,0.5),pch=c(18,20,22), col=c(3,4,5));
```

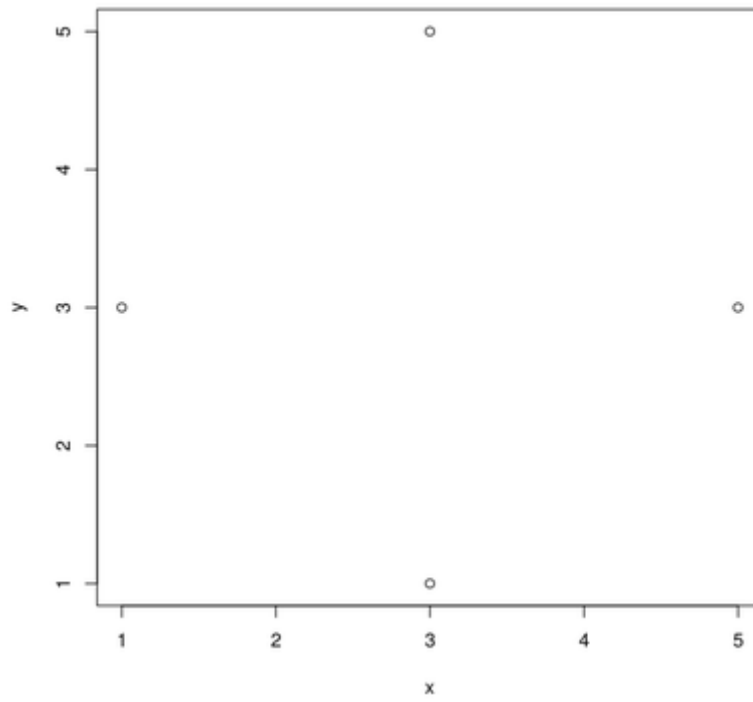


Figure 1: ImagesOfPP1/PlotTypeDefault\_resized\_60.png

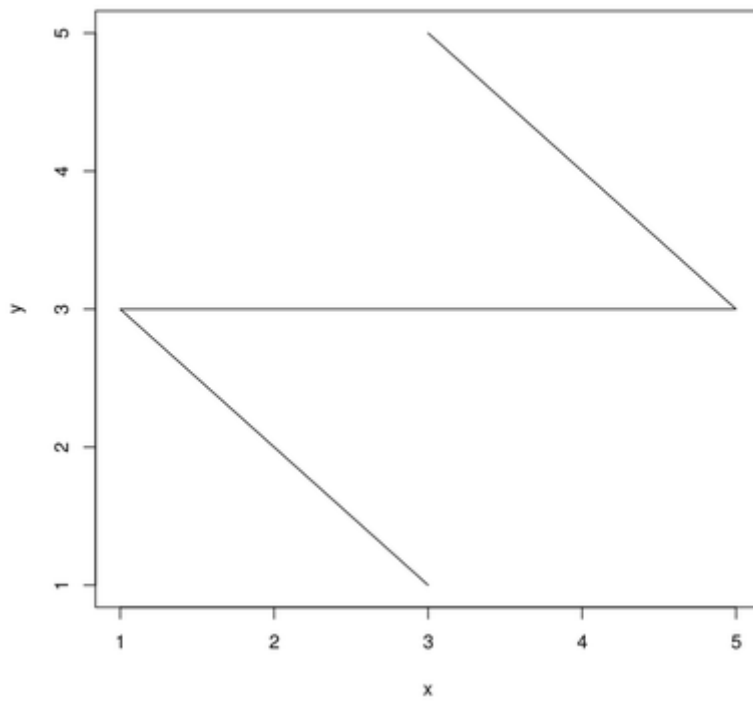


Figure 2: `plot(x,y, type="l")`

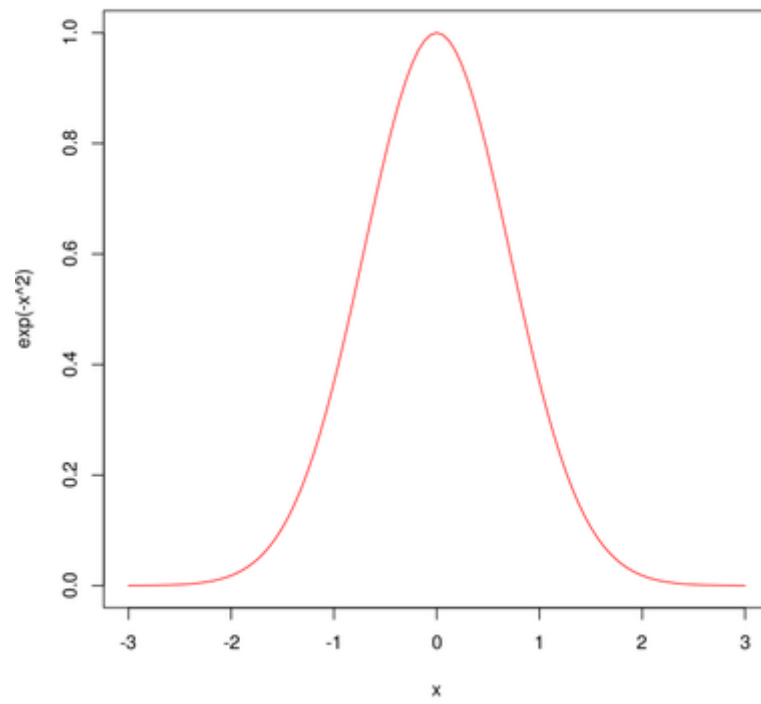


Figure 3: `plot(x,y)`

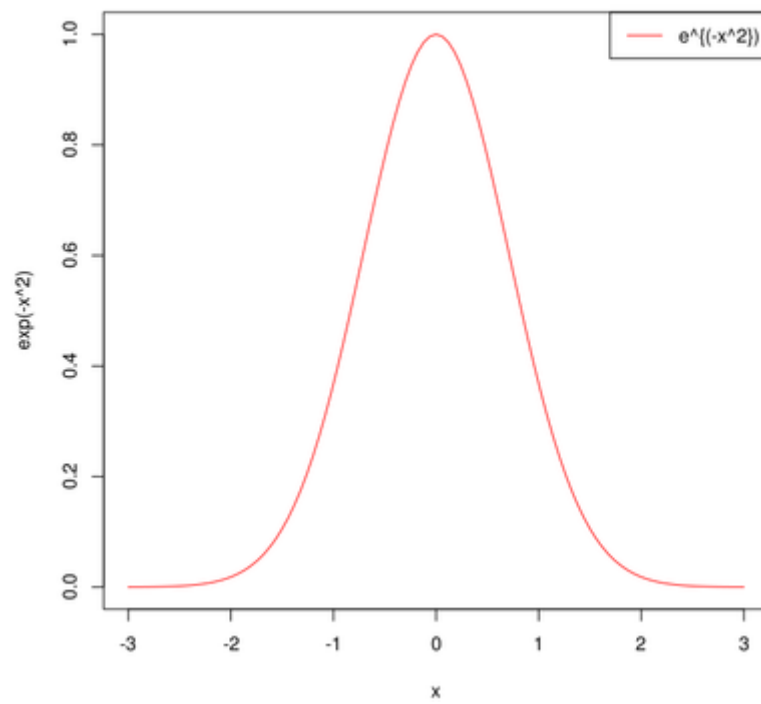


Figure 4: `plot(x,y)`

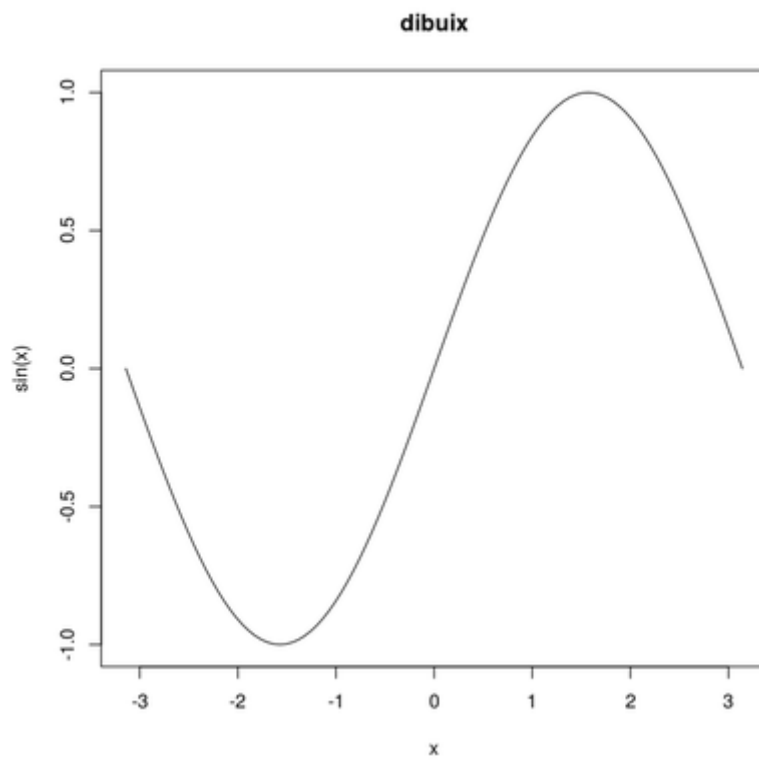


Figure 5:

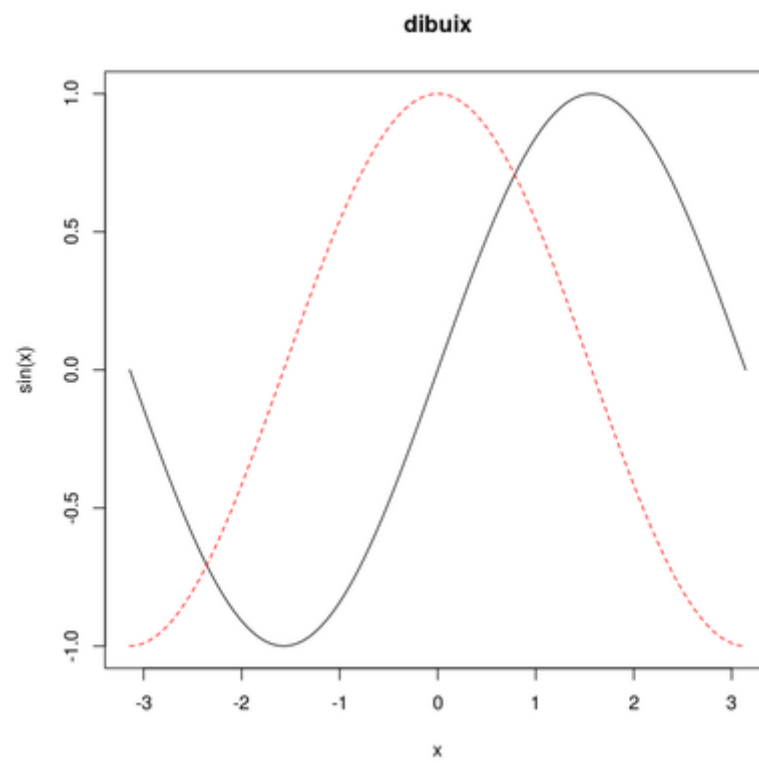


Figure 6:



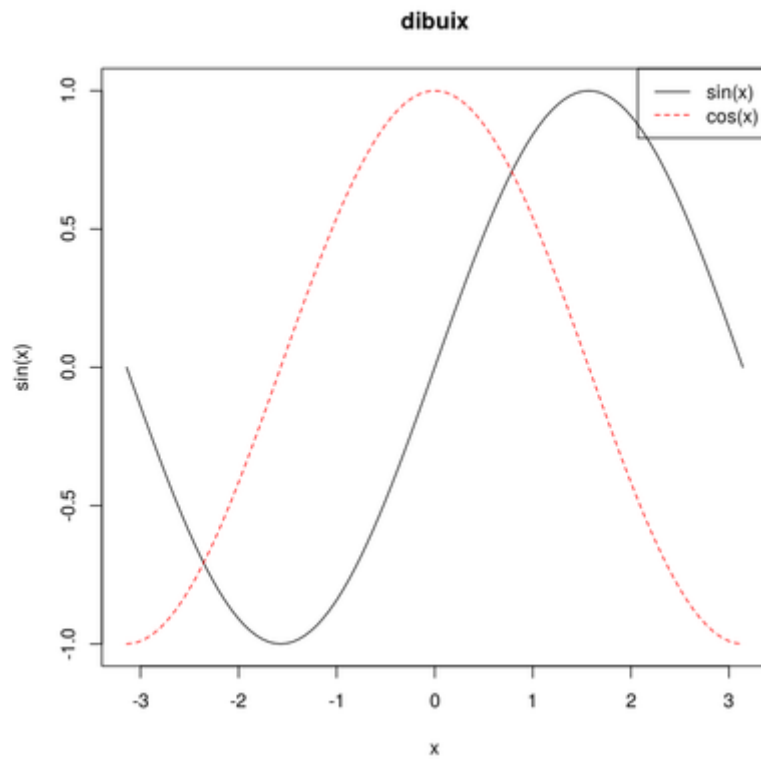


Figure 7:

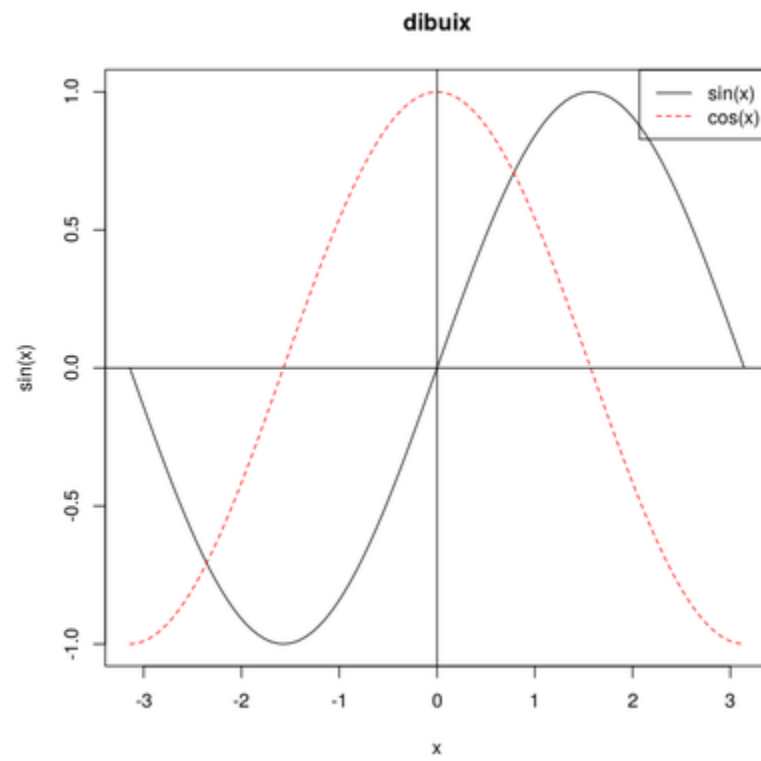


Figure 8:

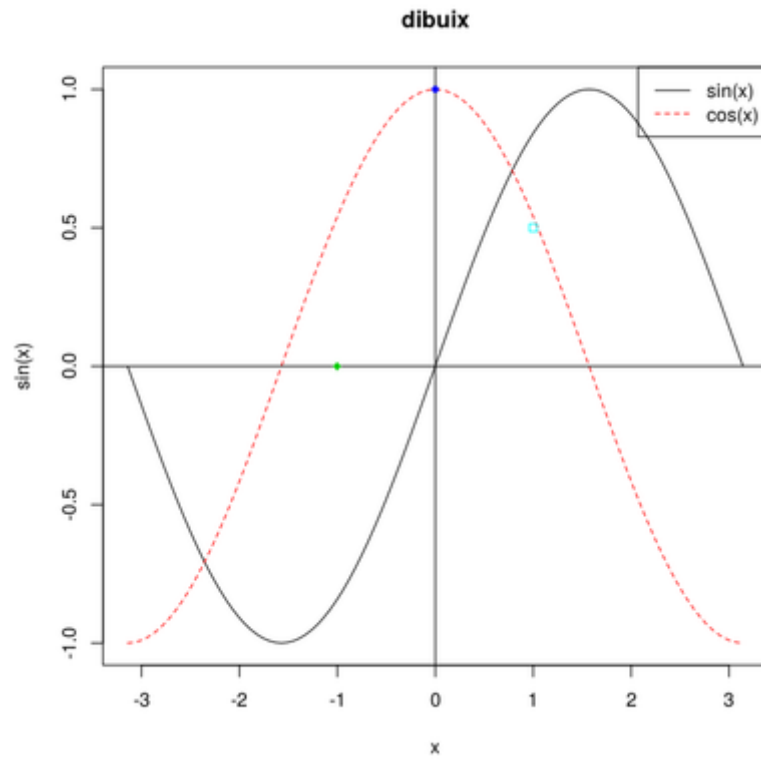
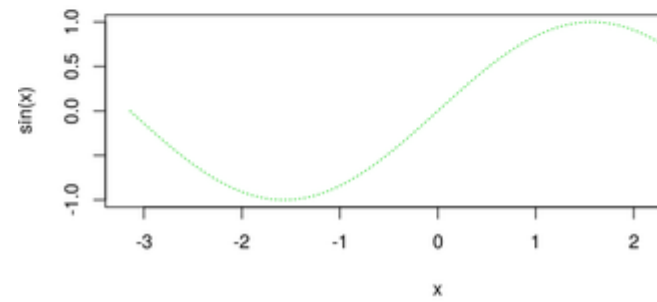
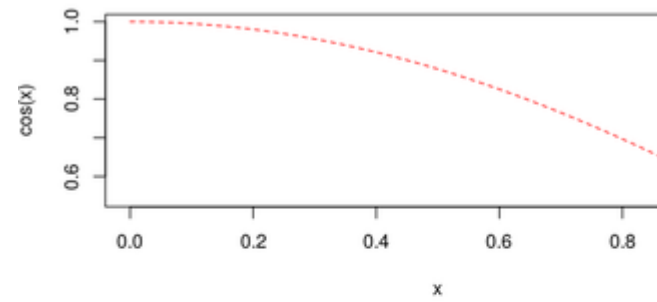
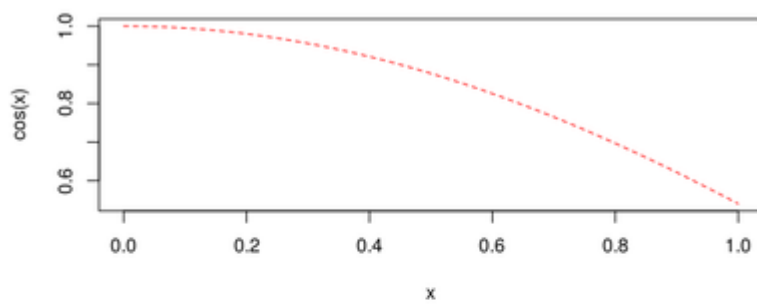


Figure 9:

```
par(mfrow=c(2,1)); curve(sin(x),-pi,pi,lty=3,col=3); curve(cos(x),lty=2,col=2);
```



```
x=seq(0,2*pi,by=0.1); plot(x,sin(x)); plot(x,sin(x), type="l");
```

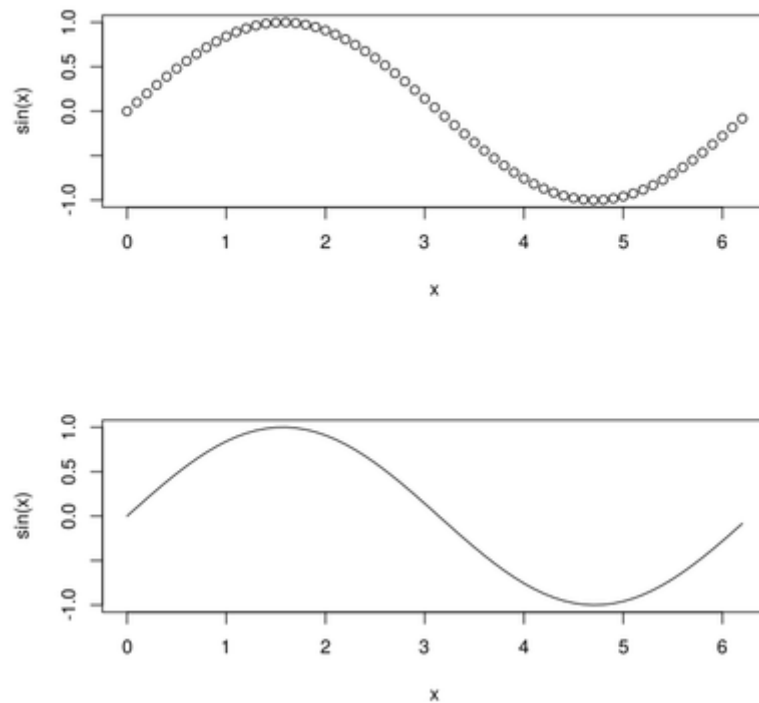


Figure 10:

**Potser rellevant dividir la taula en taules més petites per temàtica. ## Exemples ###** Exemple alçades Tenim dades de les alçades (en metres) i masses (en grams) de un grup de 6 persones. I en volem calcular L'índex de massa corporal o *IMC*, que recordem que es defineix com *el pes* (no com a força, amb Kg) dividit entre l'alçada (en metres) al quadrat. *Picant codi*:

```
# Observeu que les operacions amb vectors es fan component a component
pes=c(60,72,57,90,95,72)
alcada=c(1.75,1.80,1.65,1.90,1.74,1.91)
IMC=pes/(alcada^2)
IMC
[1] 19.59184 22.22222 20.93664 24.93075 31.37799 19.73630
```

## Exercicis

### Exercici 1

Amb els vectors  $a=c(2,1,4,7)$  i  $b=c(4,0,-1,7)$ , calculeu  $c=2*a+5*b$  i  $d=a*b$ , on aquesta darrera expressió vol dir el *producte component a component* (producte escalar de vectors).

**Solució:**

```
> a=c(2,1,4,7); b=c(4,0,-1,7); c=2*a+5*b; d=a*b; c; d;
[1] 24 2 3 49
[1] 8 0 -4 49
```

## Exercici 2

Escriuiu una funció de  $m$  i  $n$  que calculi les variacions  $\text{Var}(m, n)$ . Calculeu  $\text{Var}(365, 10)$ . Recordem que:

```
$$  
Var(m,n)=frac{m!}{(m-n)!}  
$$
```

**Solució:** Si fem servir directament l'expressió obtenim l'error Warning message: In factorial(345) : value out of range in 'gammafn'. Però també podem escriure  $\text{Var}(m,n)=\binom{m}{n} \cdot n!$ .

```
> Var=function(m,n){choose(m,n)*factorial(n)}; Var(365,10);  
[1] 3.70608e+25
```

““