Pràctica 2: Gràfiques de funcions

1 Gràfiques de funcions d'una variable

Les possibilitats gràfiques de *Mathematica* han sigut una de les causes del seu èxit. Per mitjà de *Mathematica* es poden dibuixar funcions i dades en dues i tres dimensions; produir gràfics de nivell i de densitat, a més a més de dibuixar objectes i figures arbitràries. En aquesta secció explorarem algunes de les possibilitats que *Mathematica* ofereix a l'hora de fer gràfiques.

Per a fer una gràfica en dues dimensions d'una función d'una variable s'utilitza el comandament `Plot`. Aquest comandament necessita almenys dos arguments, una expressió *expr*, i un rang. El rang té tres parts: la variable de l'expressió, x, un valor mínim, $x_{\min}$, i un valor màxim, $x_{\max}$.

`Plot` [expr, $\{x, x_{\min}, x_{\max}\}$]

Així, la gràfica de la paràbola $y = -x^2 + 4$ és produïda per la següent instrucció:

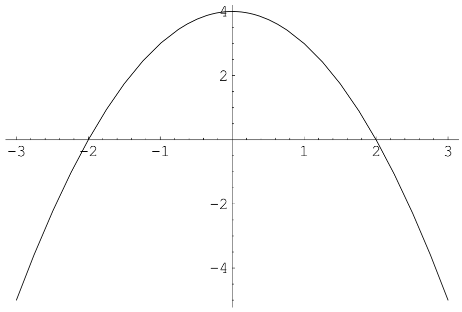
```
Plot[-x^2+4, {x,-3,3}]
```

A més a més d'aquest tipus de funcions *Mathematica* també pot fer gràfiques de funcions que tendeixen a infinit o que tenen singularitats. Així per exemple, podem produir la gràfica de la tangent en valors on la tangent es fa infinita (recordeu que els arguments de les funcions trigonomètriques han d'estar en radians, excepte que s'indique una altra cosa) utilitzant la instrucció:

```
Plot[Tan[x], {x,-2Pi,2Pi}]
```

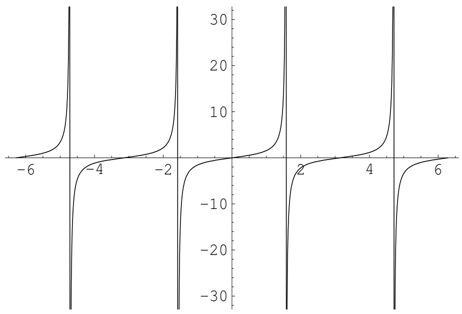
Exemple:

In[1] := `Plot [-x^2+4, {x,-3,3}]`



```
Out[1] = -Graphics-
```

```
In[2] := Plot [Tan[x], {x,-2Pi,2Pi}]
```




```
Out[2] = -Graphics-
```

Fixeu-vos que *Mathematica* no mostra tot el rang de valors de y, mostra la regió on la funció és interessant. Fora del rang en què està dibuixada la funció el comportament de la tangent no és especialment rellevant.

2 Opcions

Quan *Mathematica* dibuixa una gràfica, ha de prendre moltes decisions. Aquestes decisions poden ser modificades depenent dels valors de les opcions. Per mitjà de `??Plot` podem veure totes les opcions per a la funció `Plot` i els seus valors per defecte.

Aquestes opcions es poden especificar en qualsevol instrucció després dels arguments requerits:

`Plot` [expr, $\{x, x_{\min}, x_{\max}\}$, opcions]

Les opcions s'especifiquen donant el nom de l'opció conjuntament amb el valor; per exemple, si volem determinar el rang dels valors de y usaríem l'opció `PlotRange`. Finalment, si no s'especifica una determinada opció s'usa el seu valor per defecte.

En usar aquestes opcions s'han de tindre en compte els objectius que es persegueixen ja que algunes vegades és interessant conèixer tots els possibles valors d'una funció; en canvi, altres vegades, usar tots aquests valors dóna un gràfic que no ajuda molt a la seua avaluació.

3 Modificació de l'estil d'una gràfica

Per mitjà de l'opció `PlotStyle` es pot canviar el gruix, color i estil d'una corba. El gruix es canvia utilitzant l'expressió `Thickness`, l'argument del qual [a] és la raó de l'ample de línia al de tot el gràfic (El valor inicial per a la funció `Plot` és `Thickness[0.004]`).

```
Plot[expr, rang, PlotStyle->Thickness[a]]
```

La funció `RGBColor` permet especificar-hi un color. Aquesta funció té tres arguments: el primer és la quantitat de roig (Red), el segon de verd (Green) i el tercer de blau (Blue). Aquests arguments han de ser un nombre entre 0 i 1, on 1 indica la presència del color i 0 la seua absència. Es pot utilitzar també `GrayLevel` si es desitja un ombreig gris; el seu argument és un nombre entre 0 i 1.

`Plot`[expr, rang, `PlotStyle`− >`RGBColor[roig,verd,blau]]`
`Plot`[expr, rang, `PlotStyle`− >`GrayLevel[g]]`

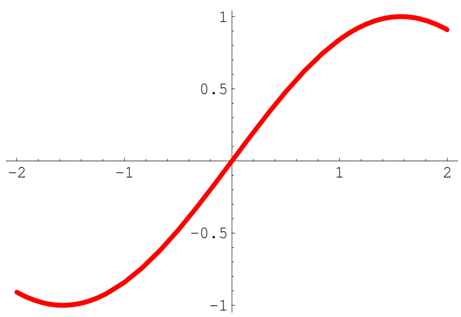
La funció `Dashing` crea una línia a traços en què els successius traços dibuixats o no dibuixats són de longitud d_1,d_2,..., els arguments de la funció. Les longituds d_1,d_2,... s'especifiquen com a fraccions de l'ample total del gráfic.

`Plot`[expr, rang, `PlotStyle`− > `Dashing`[{d_1, d_2, ...}]]

Totes aquestes opcions també poden usar-se simultaniament amb `PlotStyle` si s'especifiquen en una llista de llistes. La qual cosa permet dibuixar diverses corbes al mateix temps amb estils diferents.

Per exemple, si volem dibuixar la funció $\sin(x)$ en l'interval $[-2, 2]$ de color roig i amb un gruix que represente 0.01 respecte del total de la gràfica, utilitzaríem l'ordre:

```
In[3] := Plot[Sin[x],{x,-2,2},
         PlotStyle->{RGBColor[1,0,0],Thickness[0.01]}]
```



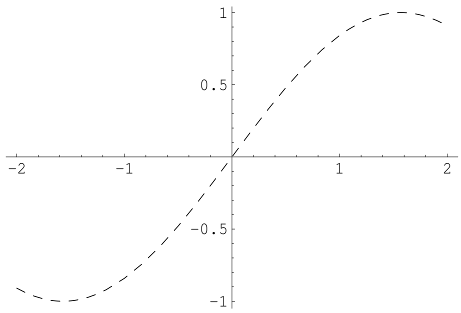

```
Out[3] = -Graphics-
```

En la gràfica anterior, la funció $\sin(x)$ seria de color roig.

Si volem dibuixar la funció $\sin(x)$ en l'interval $[-2, 2]$ amb línies discontínues on cada ratlla sigà de longitut igual a 0.02 respecte de la mida total de la gràfica, utilitzaríem l'ordre:

In[4] := `Plot[Sin[x],{x,-2,2},`

`PlotStyle->Dashing[{0.02}]]`



Out[4] `= -Graphics-`

4 Gràfiques de diverses funcions

`Plot` permet obtindre la gràfica de diverses funcions al mateix temps, per a això s'han de escriure les equacions de les distintes corbes en forma de llista en el primer argument de `Plot`

`Plot`[{funció 1, funció 2, ...}, $\{x, x_{\min}, x_{\max}\}$].

5 Punts de tall entre dues gràfiques

Per calcular les abcisses del punts de tall de les gràfiques de dues funcions $f(x)$ i $g(x)$ podem utilitzar les instruccions `Solve` o `NSolve`:

`Solve[f[x]==g[x],x]` ens calcula les abcisses dels punts de tall utilitzant mètodes algebraics de resolució d'equacions (**sempre i quan es puga fer**).

`NSolve[f[x]==g[x],x]` ens calcula el mateix però utilitzant mètodes numèrics de resolució d'equacions. Ens dóna directament aproximacions decimals de les solucions. Si volem especificar el nombre de xifres significatives exactes que volem, podem escriure

`NSolve[f[x]==g[x],x,WorkingPrecision->n]`,

on n es el nombre de xifres.

A vegades, *Mathematica* ens diu que no pot aproximar les solucions de l'equació amb la instrucció `NSolve[f[x]==g[x],x]`. Açò és degut a que els mètodes numèrics que utilitza són vàlids per a calcular TOTES les solucions, tant reals com complexes no reals. Escrivint `NSolve[f[x]==g[x],x,Reals]`, utilitzarà uns altres mètodes que són **específics** per trobar solucions reals i, aleshores, és molt probable que sí que ens puga aproximar les solucions.