

Resumo dos Comandos

- Lista de comandos a ser tratada

- `Plot`
- `Plot3D`
- `ContourPlot`
- `ParametricPlot`
- `StreamPlot`
- `VectorPlot`
- `Show`

1 `Plot`

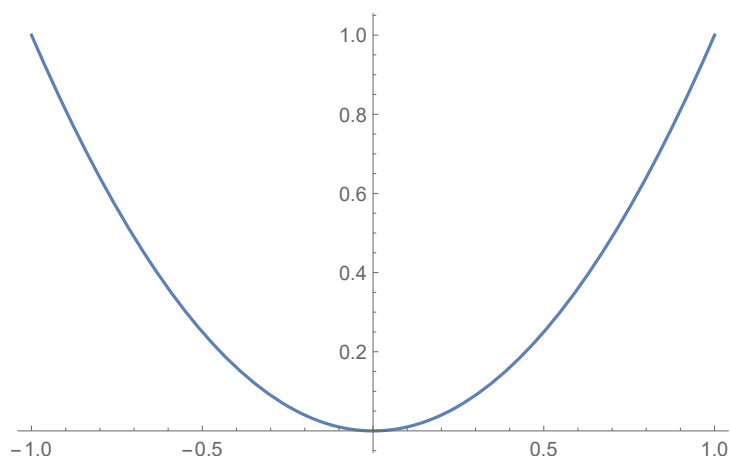
Normal recebe uma função que depende de somente uma variável seguida da definição do intervalo de amostragem desejado. Então imagine que se deseja plotar a função do segundo $f(x) = x^2$, para fazer isso basta tomar o lado direito da função e utilizar como primeiro parâmetro do `Plot` como segue.

```
1 Plot[x^2]
```

Porém, não é adequado parar por aqui, é necessário definir a variação de x em uma dimensão que vai abranger o “teatro” escolhido. Dessa maneira, imagine que queiramos plotar essa função de modo a abranger os valores de x que vão de -1 a 1, então aplicamos `{x, -1, 1}`. Como consequência, percebe-se que o programa lança valores de x abrangendo o domínio explicitado de modo a conformar a representação gráfica no menor espaço possível de visualização do gráfico para os valores de x propostos

```
1 Plot[x^2, {x, -1, 1}]
```

Da forma como está, se dermos **enter** o comando será rodado e vamos obter



Repare que sendo $x = -1$ ou $x = 1$, $f(x) = 1$, logo o máximo valor de f mostrado é 1, não precisamos a variação de y , até mesmo por que o `Plot` não aceitaria tal sintaxe.

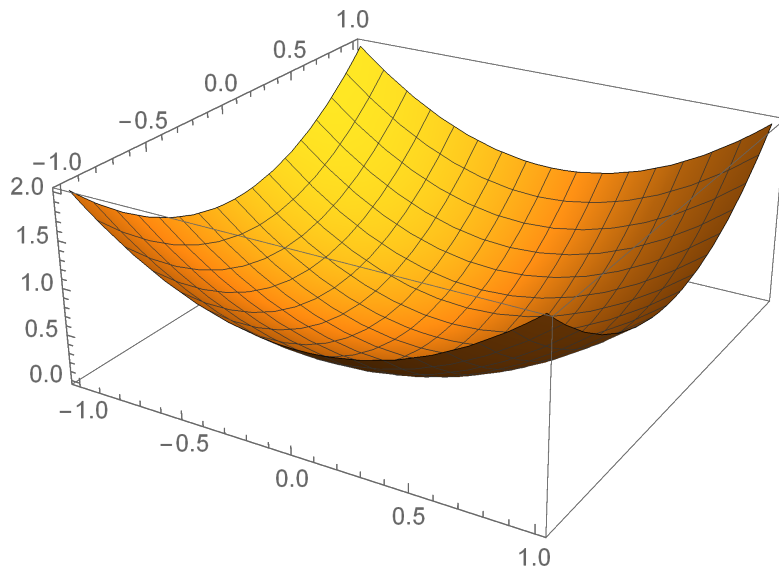
2 Plot3D

Podemos interpretar esse comando como expansão do `Plot`, porém em três dimensões. Sabendo que um ponto no espaço agora depende de x e y , consequentemente a função $f(x)$ em duas dimensões passará a ser $f(x, y)$. De forma análoga ao que foi feito para o comando anterior, imagine que queiramos plotar um paraboloide no espaço que obedeça

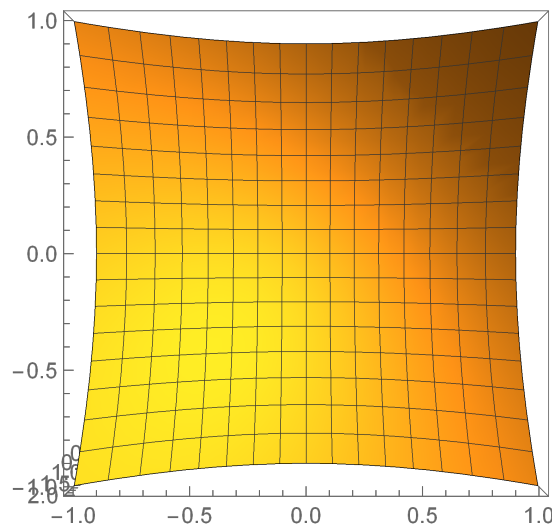
$$f(x, y) = x^2 + y^2 \quad (1)$$

Numa abordagem rápida, podemos imaginar tal superfície como a rotação de uma parábola tal como é visto em 2 em torno do eixo z no espaço, então sabendo que dependemos de mais uma variável será preciso acrescentar outro parâmetro ao `Plot3D` como segue

```
1 Plot3D[x^2+y^2,{x,-1,1},{y,-1,1}]
```

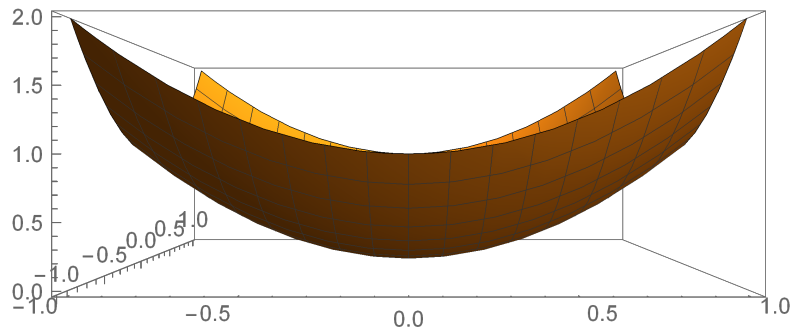


Note que se clicarmos com o botão direito do *mouse* e formos na opção *Top View* veremos



Ou seja, o intervalo de valores escolhido para x e y novamente é notado no domínio presente. Vê-se que a vista superior é um quadrado de lados valendo 2 unidades já que tanto x quanto y variam entre menos -1 e 1.

Partindo para análise em z , vamos mudar a opção de *Top View* para *Front View*. Fazendo uma análise visual vemos que a máxima altura da “caixa” que comporta o gráfico é de duas unidades. Isso ocorre pois o máximo valor que $f(x,y)$ pode assumir para o domínio escolhido ocorre quando as coordenadas são $(-1, -1)$, $(-1, 1)$, $(1, -1)$ ou $(1, 1)$ como vemos a seguir



Sabemos disso devido a simplicidade do gráfico e prevemos que a função é sempre crescente no intervalo dado.

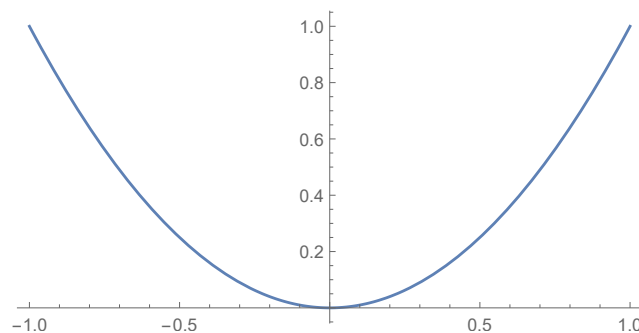
Alguns atributos de estilo podem ser passados para os Plots de modo a melhorar a visualização ou visando cumprir alguma meta de representação gráfica. Alguns dos comandos a seguir são interessantes de serem entendidos pelo menos no básico.

Para o Plot (Gráficos em 2d no geral)

- **AspectRatio**

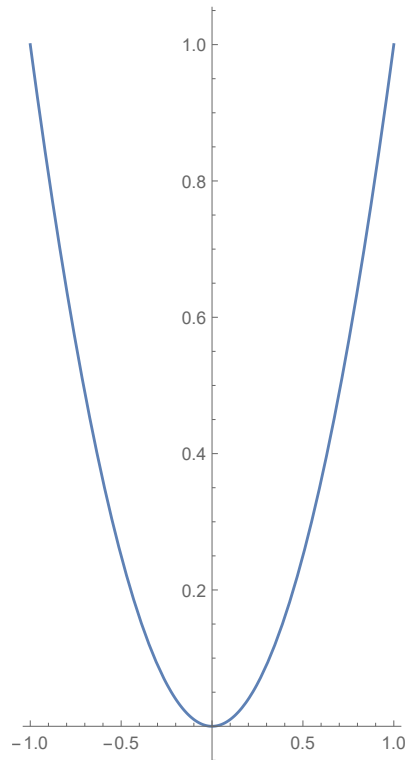
- Permite editar a proporção entre os eixos do gráfico. Imagine que quiséssemos uma proporção de 1 em x para dois em y , teríamos uma razão igual a 0.5 e podemos prever que o gráfico terá maior comprimento em x que em y como vemos a seguir com base nas linhas de comando

```
1 Plot[x^2,{x,-1,1},AspectRatio->.5]
```



Trocando por 2 o valor da propriedade vemos

```
1 Plot[x^2,{x,-1,1},AspectRatio->2]
```



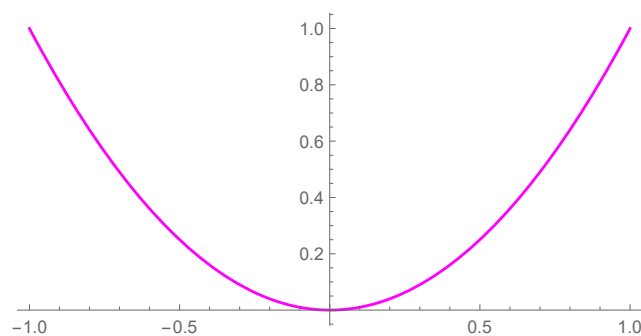
- **PlotStyle**

- Um dos atributos mais utilizados, possibilita mudar a cor do gráfico, espessura das linhas, opacidade, entre muitos outros parâmetros.

- * Mudando a cor

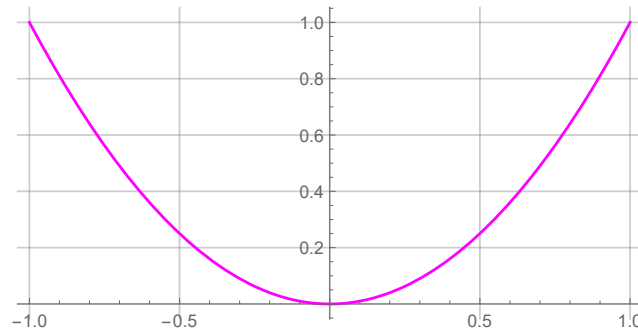
- Uma maneira aconselhável é passar diretamente o atributo de cor, como por exemplo **Magenta**, **Red**, **Blue**, **Cyan** ou utilizar o **RGBColor** para customizar com maior liberdade como segue

```
1 Plot[x^2,{x,-1,1},AspectRatio -> .5,PlotStyle -> Magenta]
```



- **GridLines**

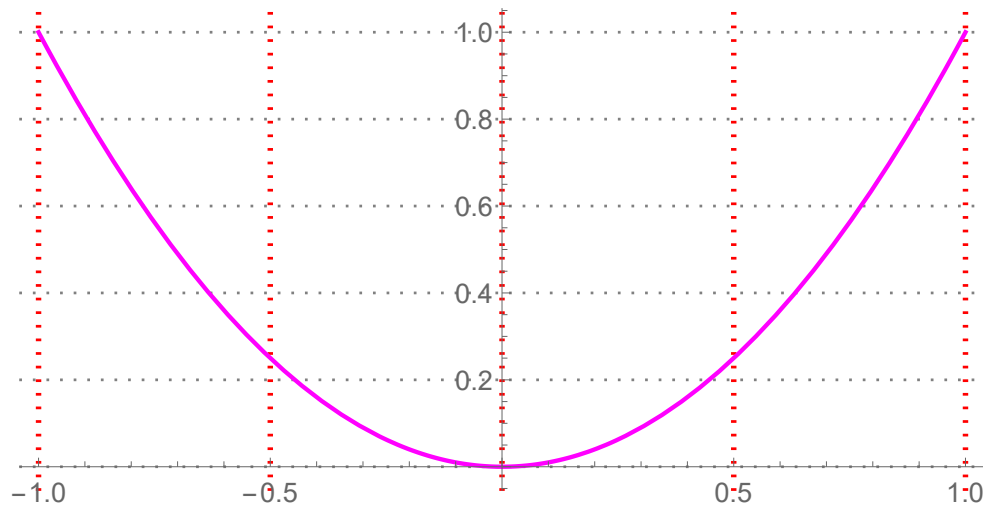
- Nos permite adicionar uma malha quadriculada ao fundo da plotagem. Esse recurso muitas vezes facilita a associação entre os eixos coordenados e nos permite ter maior precisão quantitativa para aproximações feitas visualmente. Abaixo temos alguns exemplos.



```
1 Plot[x^2,{x,-1,1},AspectRatio->.5,PlotStyle->Magenta,GridLines
->Automatic]
```

Por padrão o `GridLines` com `Automatic` renderiza linhas cheias, mas podemos alterá-las passando `GridLinesStyle->{{Dashed},{Dashed}}`. Isso fará com que tanto as linha horizontais quanto as verticais se tornem tracejadas. De forma semelhante usamos o `Dotted` para deixá-las pontilhadas. O atributo de cor também pode ser passado no interior das chaves como vemos a seguir

```
1 Plot[x^2,{x,-1,1},AspectRatio->.5,PlotStyle->Magenta,GridLines
->Automatic,GridLinesStyle->{{Red,Dotted,Thick},{Gray,Dotted}}]
```



Aumentando-se a proporção do gráfico vê-se que a primeira lista de valores contendo `Red`, `Dotted` e `Thick` tornou as linhas verticais vermelhas, pontilhadas e, como foi passado o `Thick` houve uma leve aumentada na espessura, somente para melhorar a visualização do leitor.