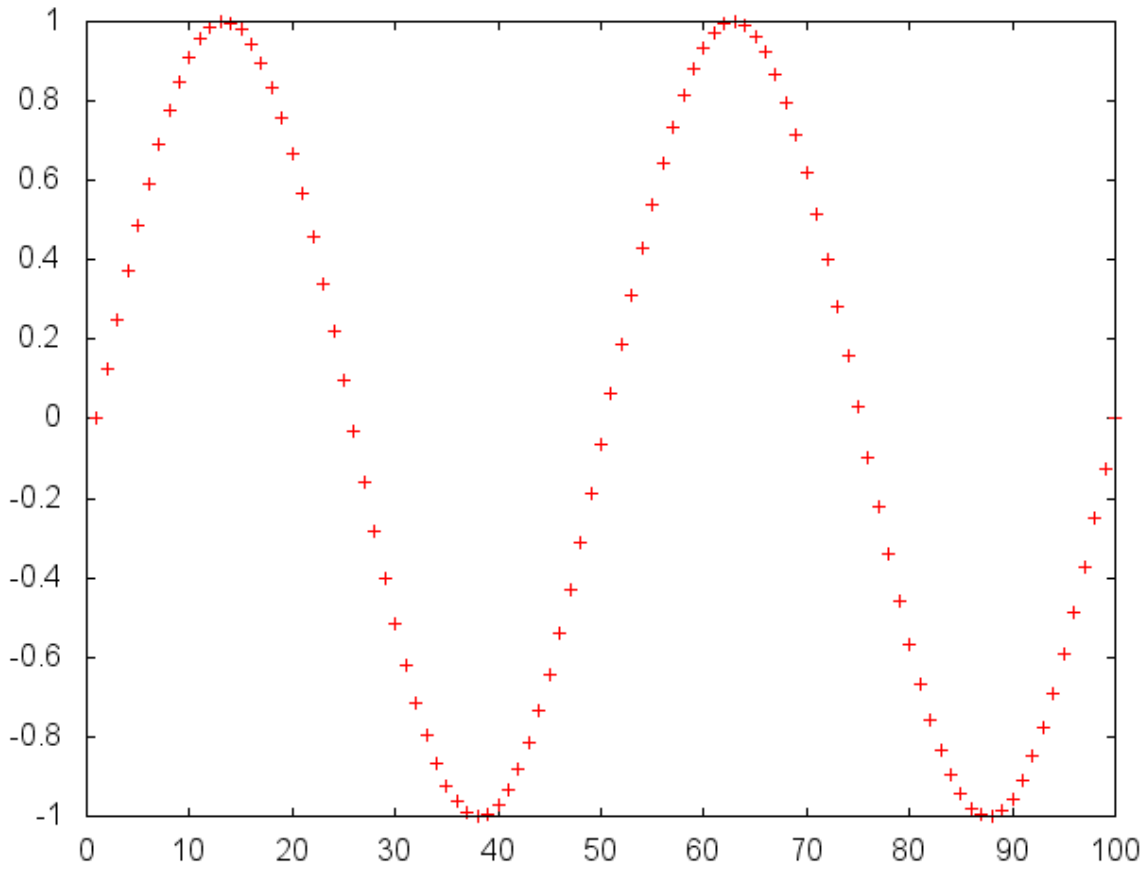
Plotting Lines

Line plotting functionality covers many configurations from simplest case of plotting a single vector to displaying multiple lines at once with custom line specifictions.

gnuplot.plot(x)

Plot vector `x` using dots of first default `Gnuplot` type.

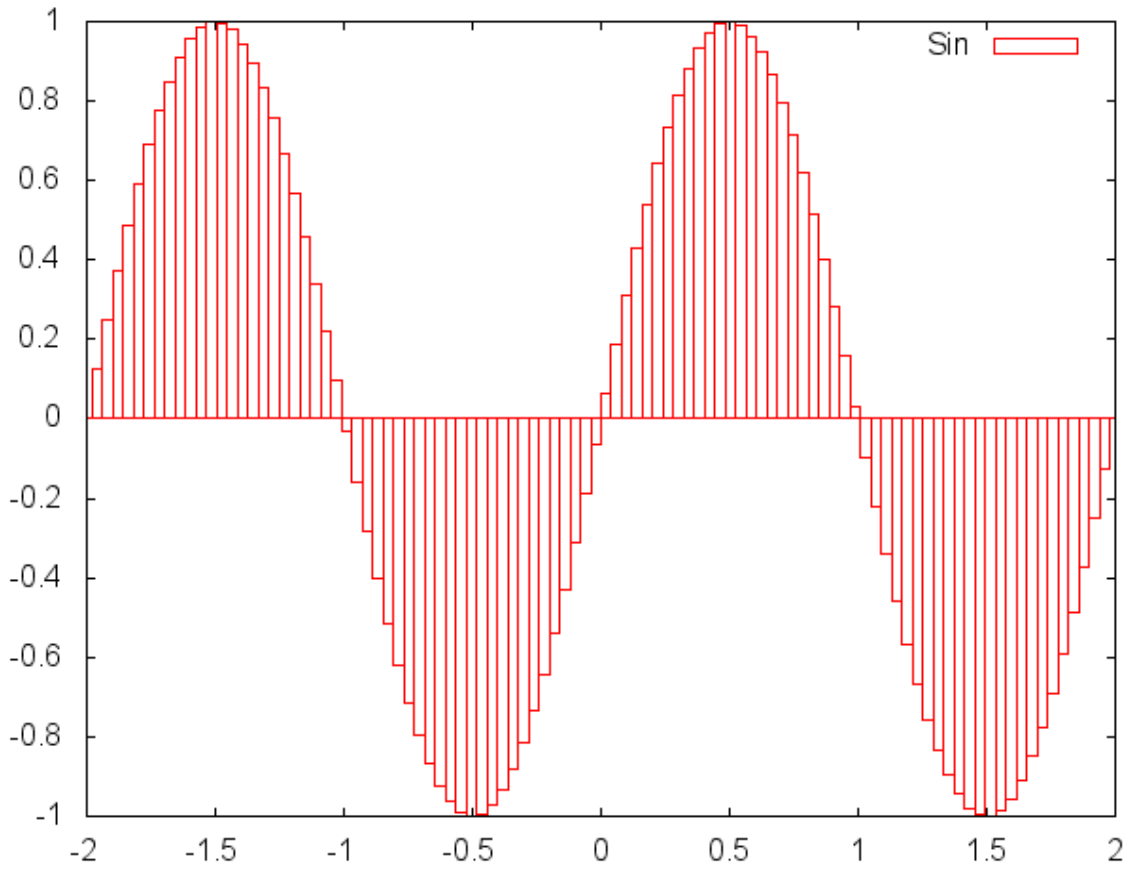
```
x=torch.linspace(-2*math.pi,2*math.pi)
gnuplot.plot(torch.sin(x))
```



In more general form, plot vector `y` vs `x` using the format specified. The possible entries of format string can be:
* `.` for dots

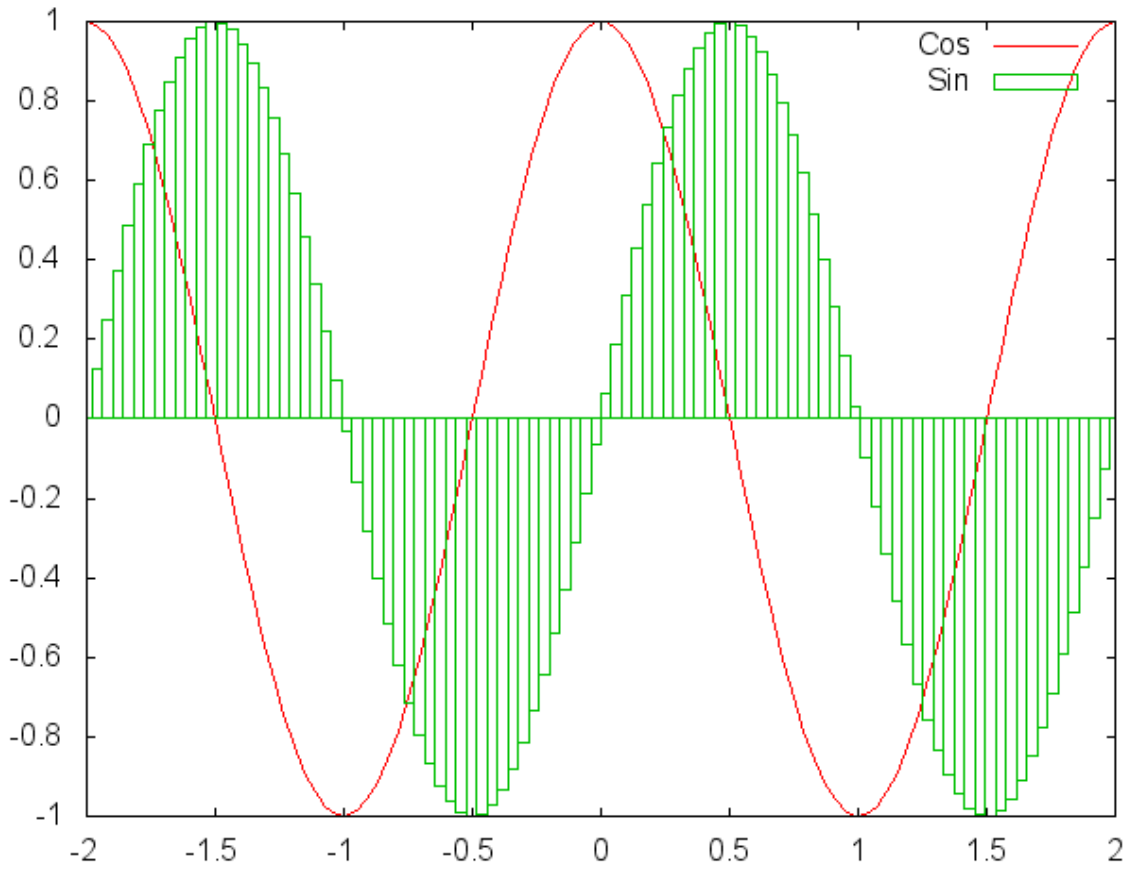
* `+` for points
* `-` for lines
* `+-` for points and lines
* `~` for using smoothed lines with cubic interpolation
* `|` for using boxes
* `v` for drawing vector fields. (In this case, `x` and `y` have to be two column vectors `(x, xdelta)`, `(y, ydelta)`)
* custom string, one can also pass custom strings to use full capability of gnuplot.

```
x = torch.linspace(-2*math.pi,2*math.pi)
gnuplot.plot('Sin',x/math.pi,torch.sin(x),'|')
```



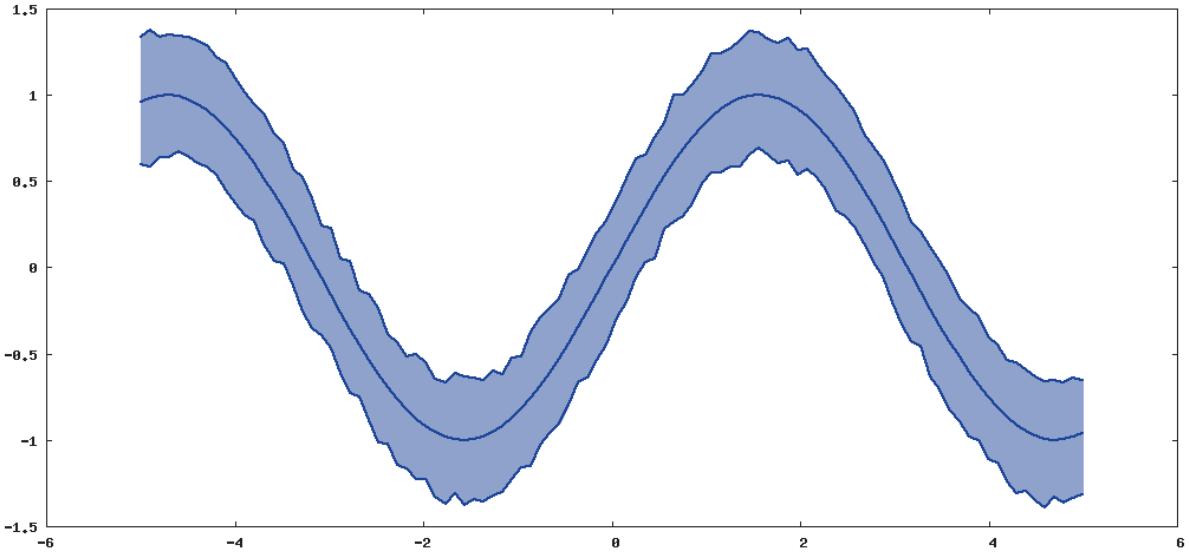
To plot multiple curves at a time, one can pass each plot struct in a table.

```
x = torch.linspace(-2*math.pi,2*math.pi)
gnuplot.plot({'Cos',x/math.pi,torch.cos(x),'~'},
{'Sin',x/math.pi,torch.sin(x),'|'})
```



One can pass data with multiple columns and use custom gnuplot style strings too. When multi-column data
is used, the first column is assumed to be the `x` values and the rest of the columns are separate `y` series.

```
x = torch.linspace(-5,5)
y = torch.sin(x)
yp = y+0.3+torch.rand(x:size())*0.1
ym = y-(torch.rand(x:size())*0.1+0.3)
yy = torch.cat(x,ym,2)
yy = torch.cat(yy,yp,2)
gnuplot.plot({yy,'with filledcurves fill transparent solid 0.5'},
{x,yp,'with lines ls 1'},{x,ym,'with lines ls 1'},{x,y,'with lines
ls 1'})
```


1.5
1
0.5
0
-0.5
-1
-1.5
-6
-4
-2
0
2
4
6