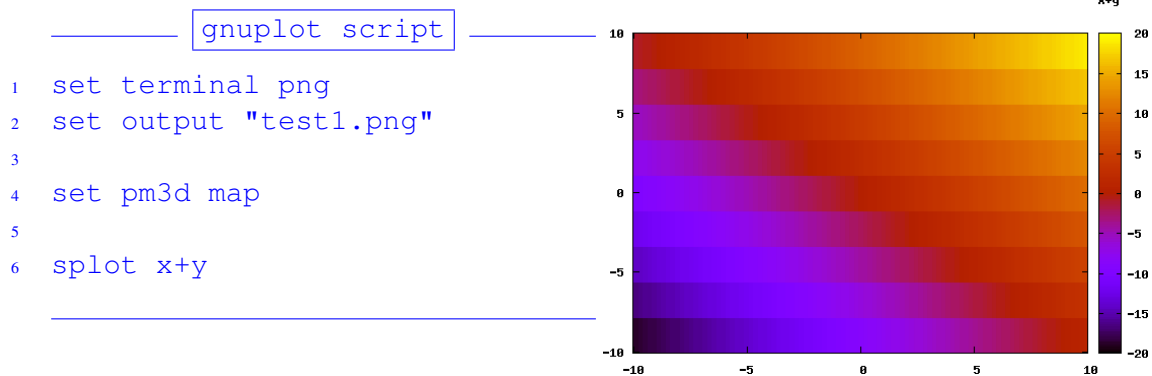


Konturplots mit Gnuplot

Um einen Konturplot mit Gnuplot erzeugen zu können muss man die Option `pm3d map` verwenden. Nun kann man mit `splot` den gewünschten Plot erstellen.

Plot einer Funktion

Im folgenden ist die Funktion $x + y$ geplottet.



Plot von Daten aus einer Datei

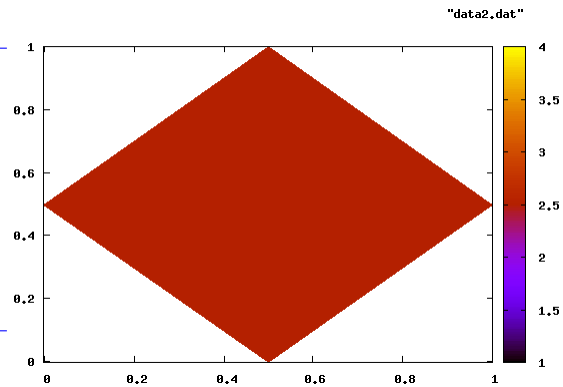
Das Plotten von Daten aus einer Datei ist nicht viel schwieriger als der Plot einer Funktion, sofern man weiß wie man die Datei formatieren muss. Die Hilfe von Gnuplot bezüglich `pm3d`, welche man mit dem Befehl `help pm3d` erhält, liefert u.a. folgendes:

... The input data come from an evaluated function or from an 'splot data file'. Each surface consists of a sequence of separate scans (isolines). The pm3d algorithm fills the region between two neighbouring points in one scan with another two points in the next scan by a gray (or color) according to z-values (or according to an additional 'color' column, see help for 'using') of these 4 corners; by default the 4 corner values are averaged, but this can be changed by the option 'corners2color'. ...

Im Prinzip liest Gnuplot also vier Punkte ein und füllt die Fläche die diese Punkte aufspannen. Dies stimmt aber leider nicht ganz, da es auf die Reihenfolge der Punkte ankommt. Im Folgenden ist ein einfaches Beispiel gezeigt. Beim Erstellen der Daten-Datei ist zu beachten, dass nach je zwei Punkte eine leere Zeile stehen muss.

gnuplot script

```
1 set terminal png
2 set output "test2.png"
3
4 set pm3d map
5
6 splot "data2.dat"
```



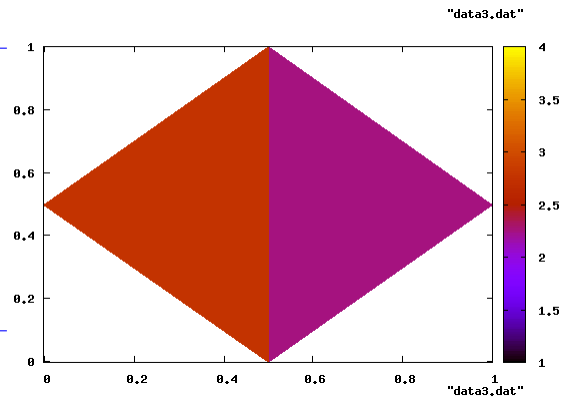
data2.dat

```
1 #x      y      z (color)
2 1      0.5    1
3 0.5    1      2
4
5 0.5    0      3
6 0      0.5    4
```

Das Problem mit der Reihenfolge tritt bei uns aber nicht auf, da wir nur Dreiecke zeichnen wollen. Um aus einem Objekt mit vier Punkten ein Dreieck zu machen, kann man nun einfach zwei der vier Punkte auf den selben Wert setzen. Gnuplot verwendet per Default für die Farbe, mit der die aufgespannte Fläche gefüllt wird, den Mittelwert der vier Ecken (Achtung: beim Dreieck wird ein Eckwert doppelt gezählt, was sich aber nicht vermeiden lässt). Wenn man nun aber statt der Option `pm3d map` die Option `pm3d map interpolate n,m` verwendet, so zeichnet Gnuplot einen Farbverlauf, bei dem die Fläche in $n \times m$ Teilflächen unterteilt wird. Mehr Informationen erhält man in der Hilfe (Befehl `help pm3d`). Im folgenden Beispiel werden zwei Dreiecke gezeichnet, und zwar einmal mit und einmal ohne *interpolate*. Dabei steht das Ergebnis jeweils neben dem entsprechenden Script. Zu beachten ist, dass man in der Daten-Datei nach jedem Dreieck zwei Zeilen leer lassen muss.

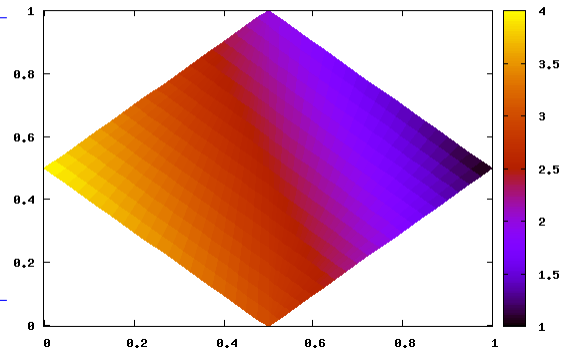
gnuplot script

```
1 set terminal png
2 set output "test3.png"
3
4 set pm3d map
5
6 splot "data3.dat"
```



gnuplot script

```
1 set terminal png
2 set output "test4.png"
3
4 set pm3d map interpolate 20,20
5
6 splot "data3.dat"
```



data3.dat

```
1 #x      y      z (color)
2 1      0.5    2
3 0.5    1      1
4
5 0.5    0      3
6 0.5    0      3
7
8
9 0.5    0      3
10 0      0.5    4
11
12 0.5    1      1
13 0.5    1      1
```