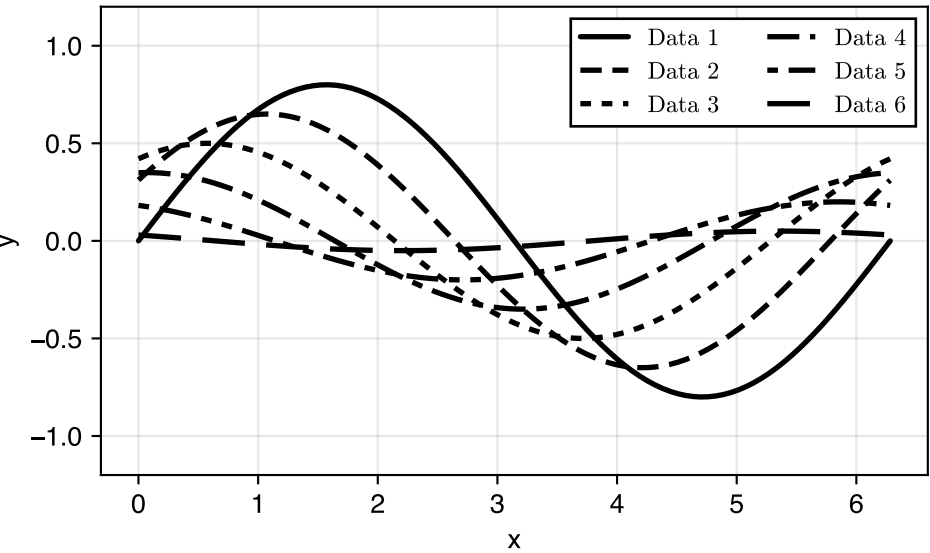


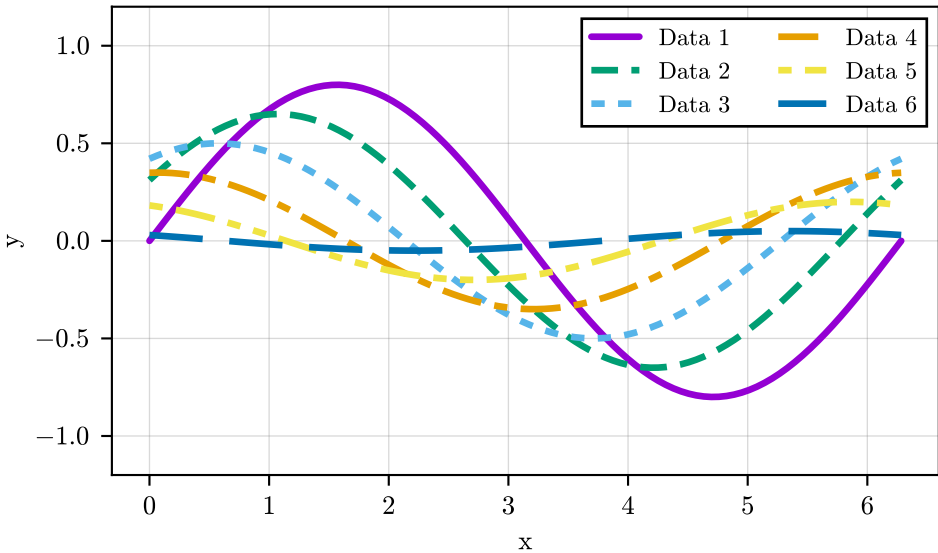
Gnuplot Style Reference - gnuplot_style.py

(with integrated gnuplot.mplstyle)

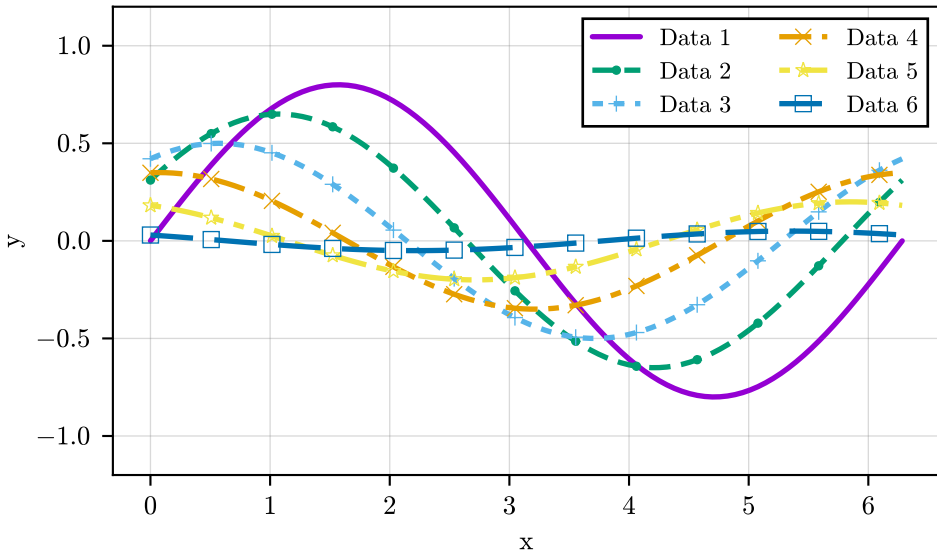
gp.use("l") - Lines Only



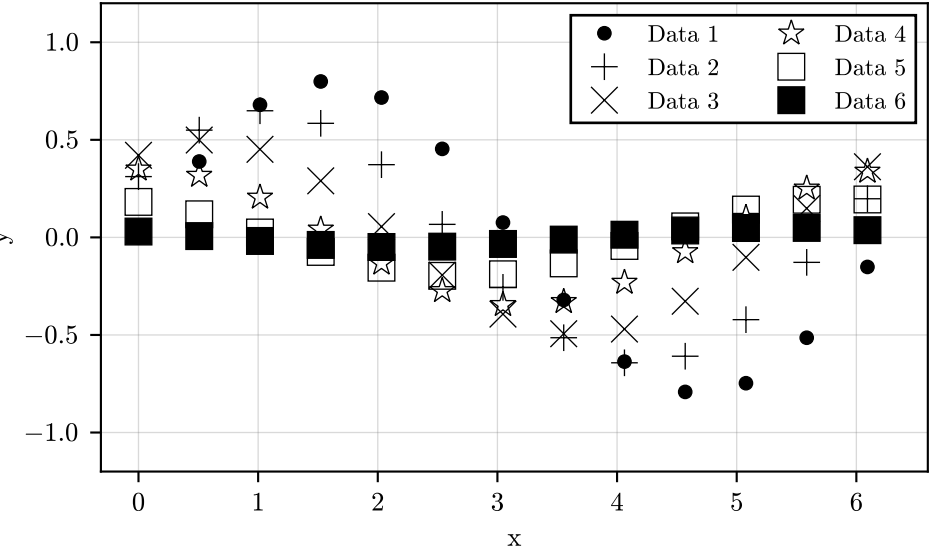
gp.use("cl") - Lines + Colors



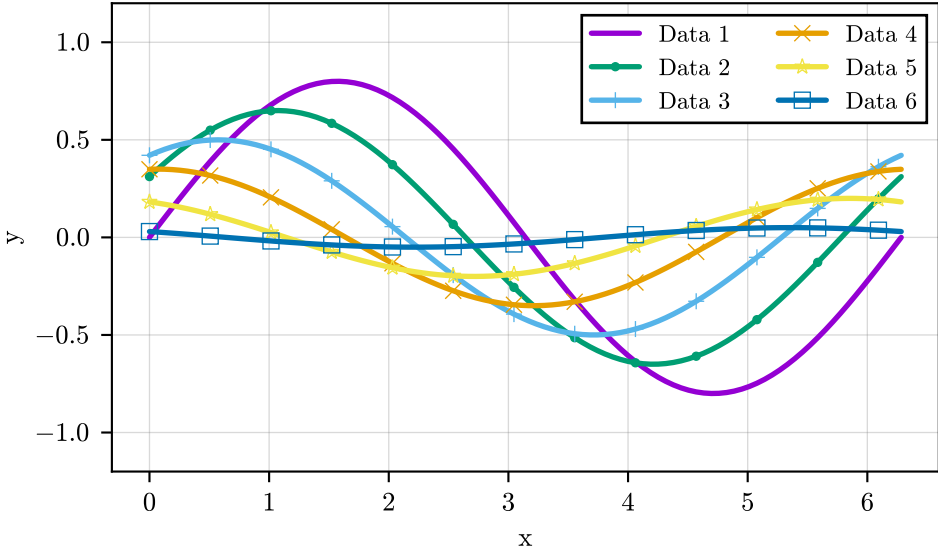
gp.use("all") - Lines + Markers + Colors



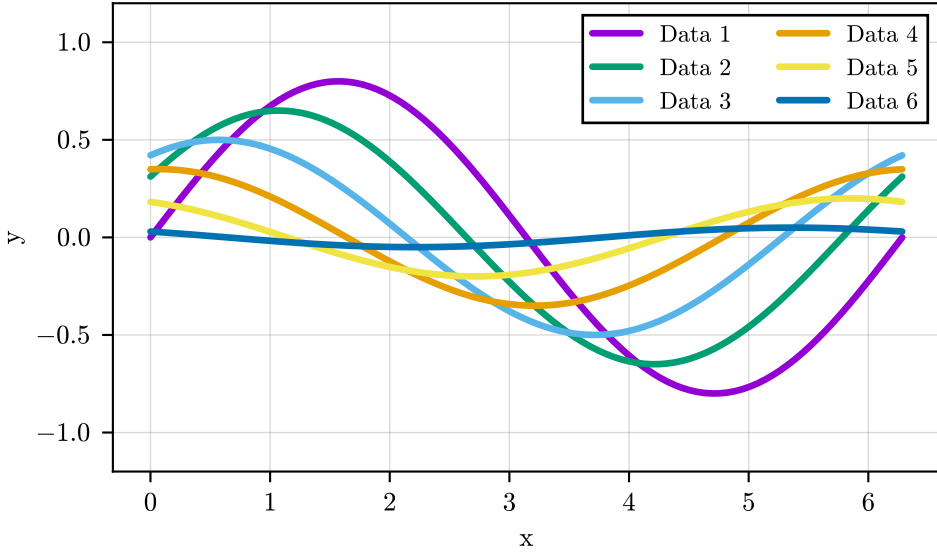
gp.use("m", skip_no_marker=True) - Markers Only



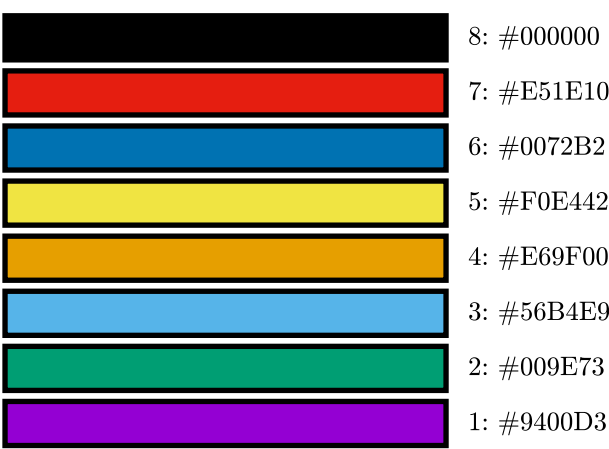
gp.use("cm") - Markers + Colors



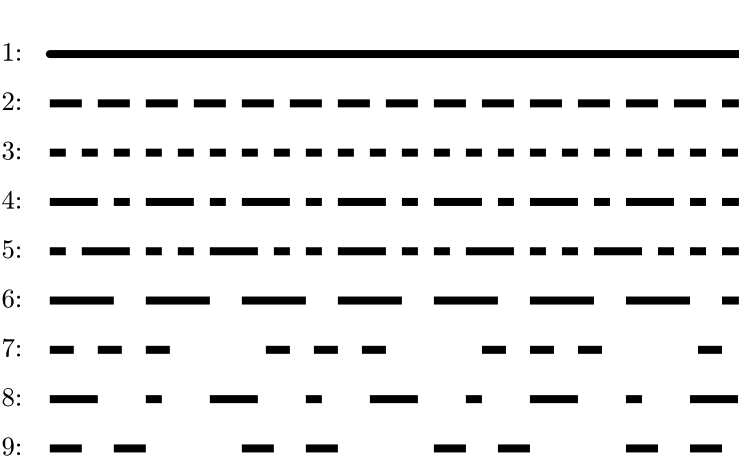
gp.use("c") - Colors Only



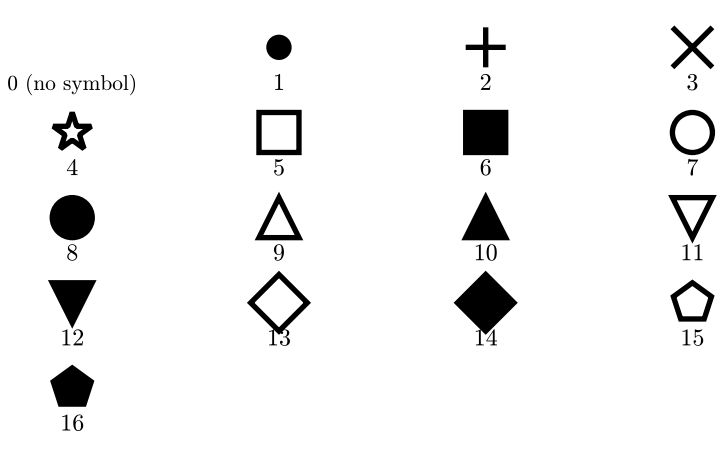
Gnuplot Colors



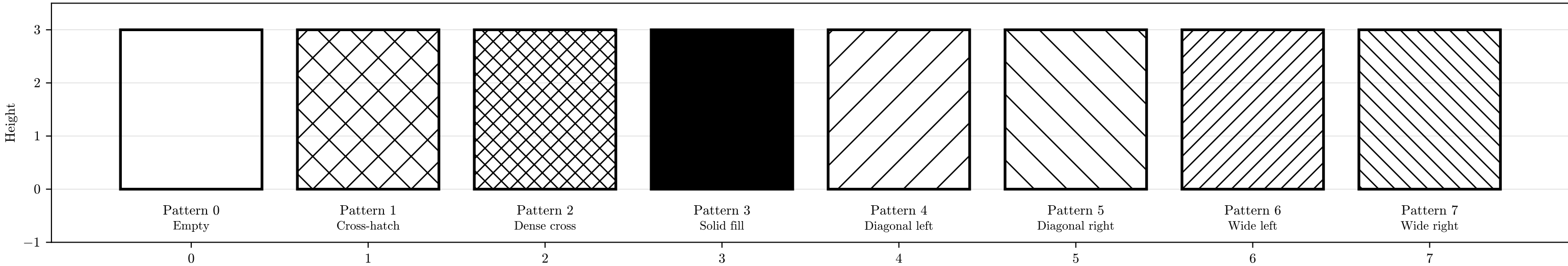
Line Styles



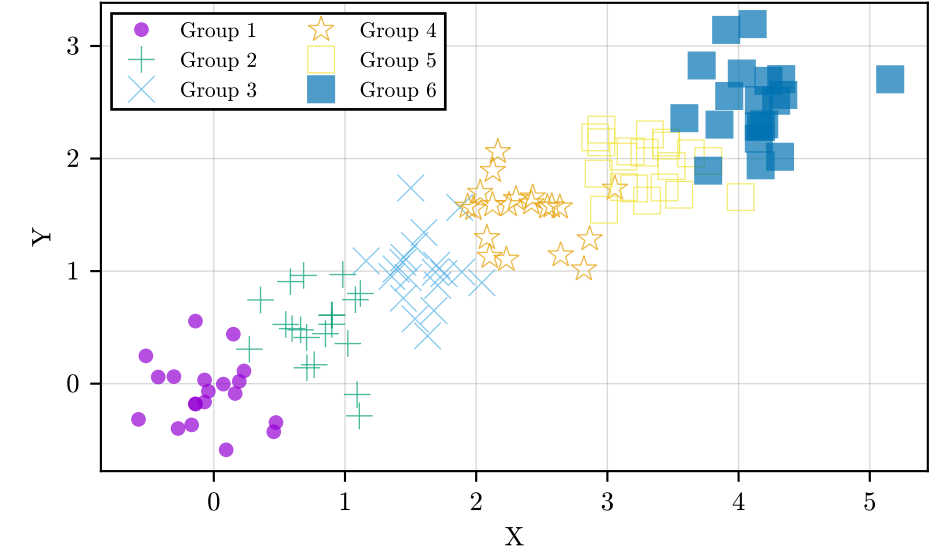
Markers (0-16)



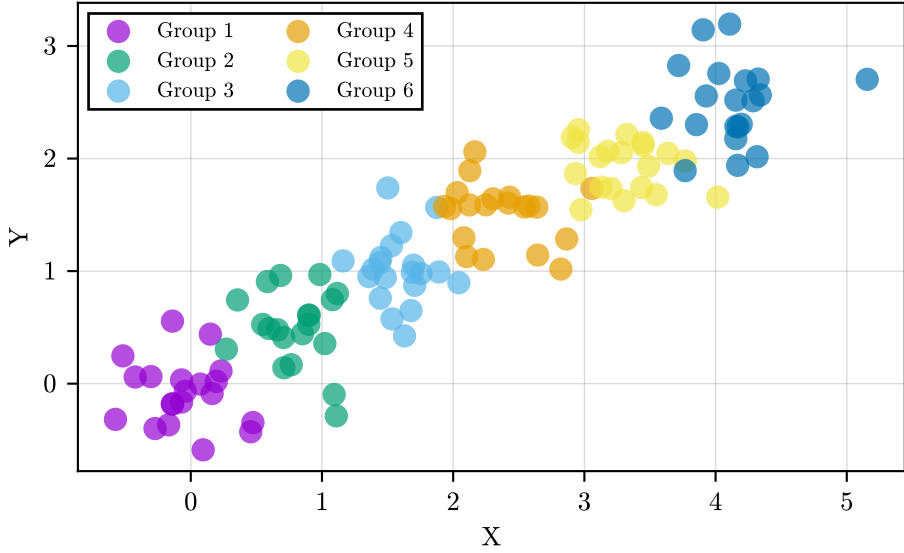
Pattern Fills for Bar Charts



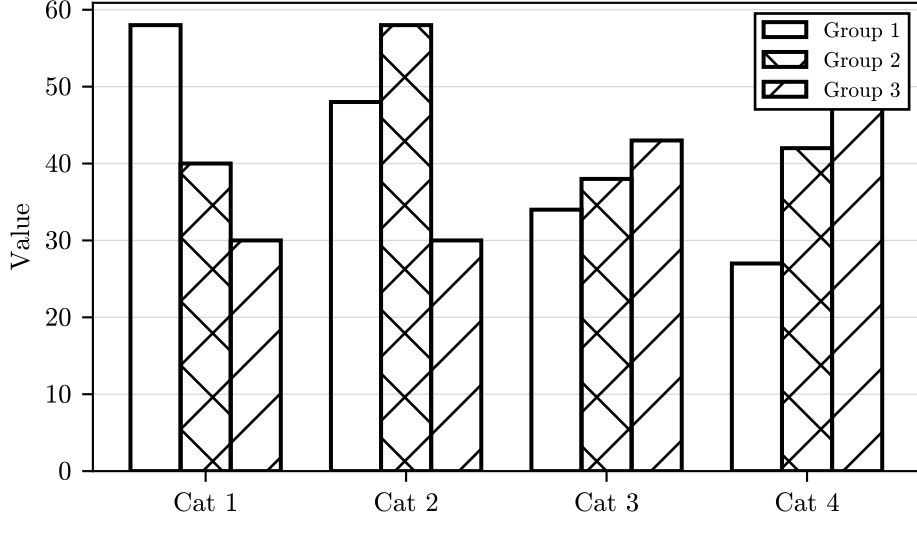
Scatter Plot (Markers + Colors)



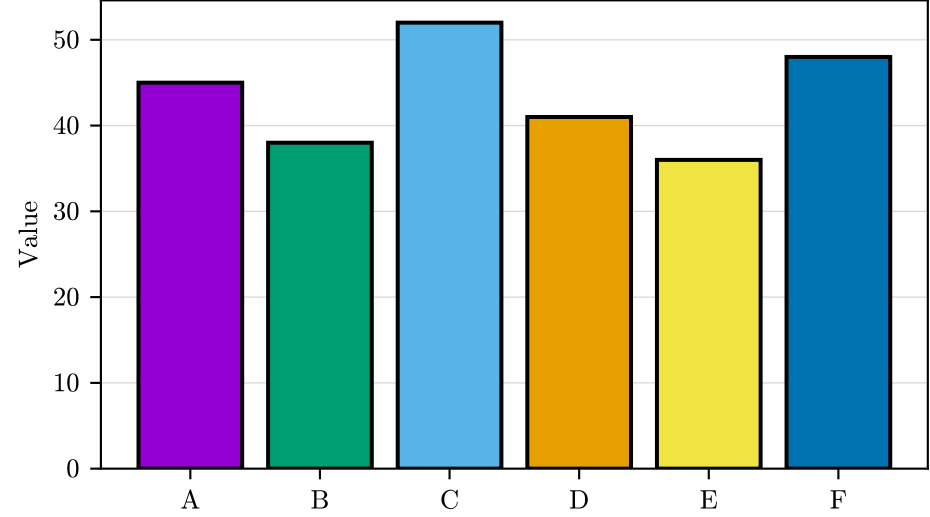
Scatter Plot (Colors Only)



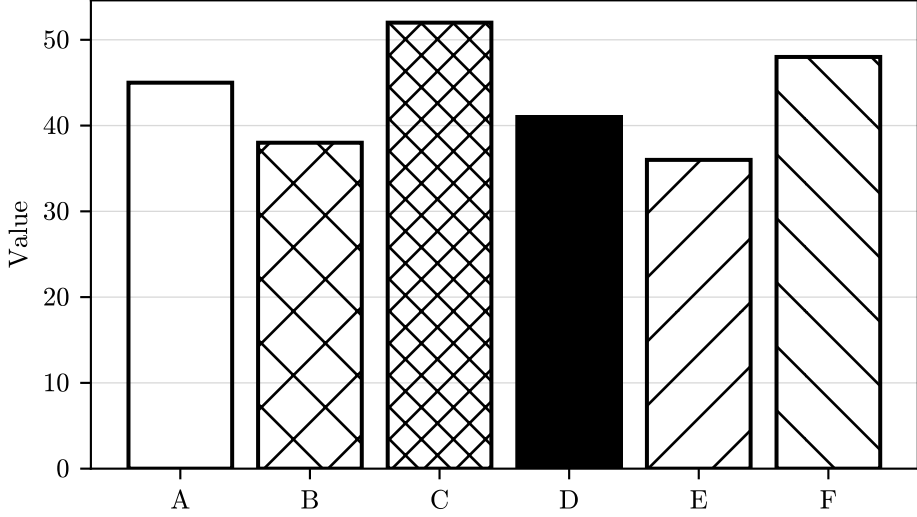
Grouped Bar Chart (Patterns)



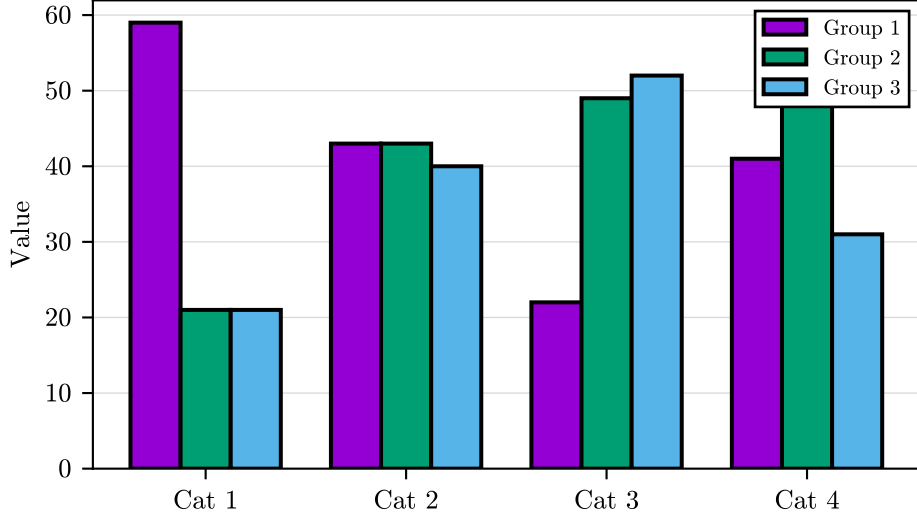
Bar Chart (Colors)



Bar Chart (Patterns)



Grouped Bar Chart (Colors)



Usage Examples

```
import gnuplot_style as gp
import matplotlib.pyplot as plt

# Apply a style (automatically loads gnuplot.mplstyle)
gp.use('cl') # Colors + Lines
gp.use('cm') # Colors + Markers
gp.use('all') # All features

# Disable mplstyle if needed
gp.use('c', apply_mplstyle=False)

# Then plot normally
plt.plot(x, y)

# For bar patterns:
bars = plt.bar(x, heights)
gp.apply_pattern(bars, pattern_index)
```

Available Styles:
'c' - Colors only
'l' - Lines only (black)
'm' - Markers only
'cl' - Colors + Lines
'cm' - Colors + Markers
'all' - All combined

Convenience functions:
gp.colors()
gp.lines()
gp.markers()
gp.colors'lines()
gp.colors'markers()
gp.all()

Integrated gnuplot.mplstyle:
Serif fonts (Computer Modern)
Figure: 6.4x4.8, 300 DPI
Ticks: gnuplot style
Legend: white bg, black border
No grid by default
Error bars: no caps
And much more...