Plotting with Matplotlib

Computational Design Laboratory

Department of Aerospace Engineering
Iowa State University

September 7, 2020

Outline

- Installing Matplotlib
- Matplotlib examples
- Further reading

Installing Matplotlib

- Linux: open terminal and type pip install matplotlib
- Windows: open Anaconda prompt from start menu and type 'pip install matplotlib' (always install packages through Anaonda prompt)

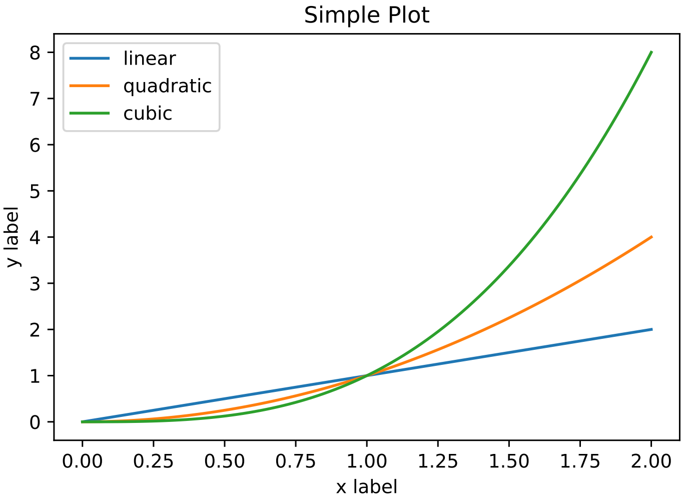
1D plot

- Open the folder you have downloaded from GitHub
- Open the 05_plotting_with_matplotlib folder
- Open 1d_plot.py from Spyder and run it
- This code can also be found here : ▸ Link

This tutorial covers:

- Creating a 1D plot
- Labeling the x and y axes
- Including a title in the plot
- Including the legend in the plot

1D plot


Simple Plot
linear
quadratic
cubic
y label
x label
0
1
2
3
4
5
6
7
8
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

Contour plot

- Open the folder you have downloaded from GitHub
- Open the 05_plotting_with_matplotlib folder
- Open contour.py from Spyder and run it
- This code can also be found here : ▸ Link

This tutorial covers:

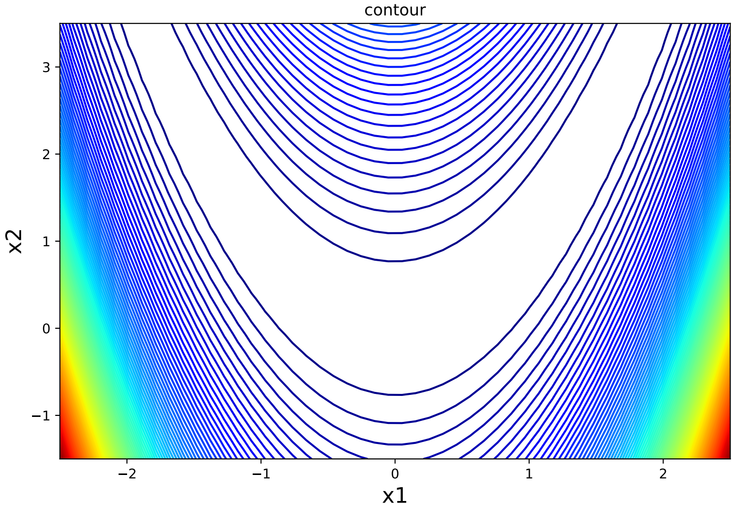
- Creating a 2D contour plot
- Defining the number of contour lines
- Labeling the x and y axes
- Including a title in the plot
- Including the legend in the plot
- Saving the plot

Contour plot

The function plotted is:

$$f(\mathbf{x}) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2$$

Contour plot



3D contour plot

- Open the folder you have downloaded from GitHub
- Open the 05_plotting_with_matplotlib folder
- Open contour_3d.py from Spyder and run it
- This code can also be found here : ▸ Link

This tutorial covers:

- Creating a 3D contour plot
- Defining the orientation of the plot
- Labeling the x and y axes
- Including a title in the plot
- Including the legend in the plot
- Saving the plot

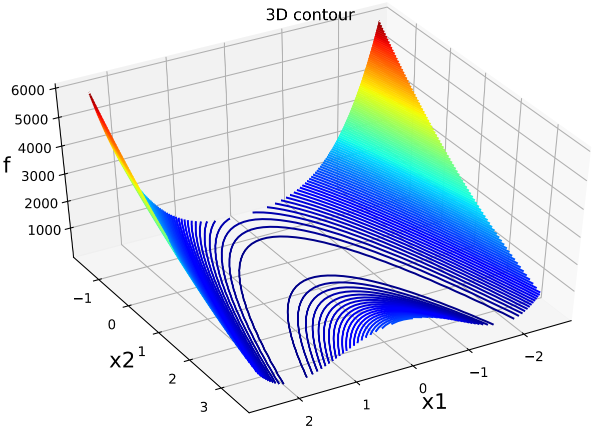
3D contour plot

The problem is defined as follows:

$$\min_{\mathbf{x}} f(\mathbf{x}) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2$$

The true minimum is $\mathbf{x} = (1, 1)$

3D contour plot



Optimization path plot

- Open the folder you have downloaded from GitHub
- Open the 05_plotting_with_matplotlib folder
- Open opt_path.py from Spyder and run it
- This code can also be found here : ▸ Link

This tutorial covers:

- Running a gradient-based optimizer
- Saving the intermediate iteration values
- Plotting the contour of the objective function
- Plotting the path taken by the optimizer
- Saving the plot

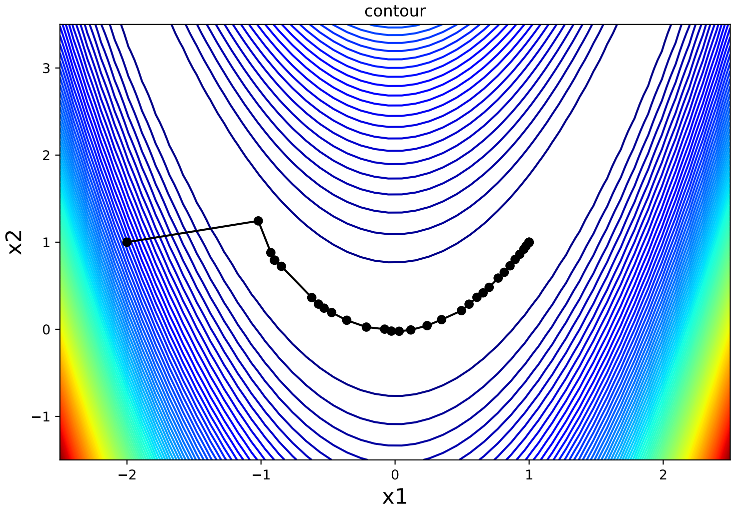
Optimization path plot

The problem is defined as follows:

$$\min_{\mathbf{x}} f(\mathbf{x}) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2$$

The true minimum is $\mathbf{x} = (1, 1)$

Optimization path plot



Convergence plot

- Open the folder you have downloaded from GitHub
- Open the 05_plotting_with_matplotlib folder
- Open conv_history.py from Spyder and run it
- This code can also be found here : ▸ Link

This tutorial covers:

- Running a gradient-based optimizer
- Saving the intermediate iteration values
- Plotting the convergence of the objective function
- Plotting the convergence of the change in design variables
- Saving the plot

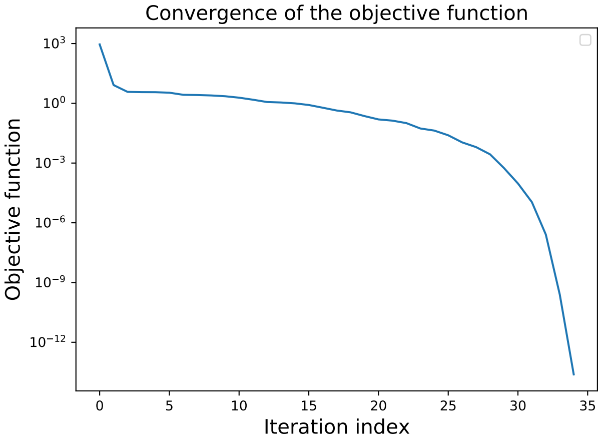
Convergence plot

The problem is defined as follows:

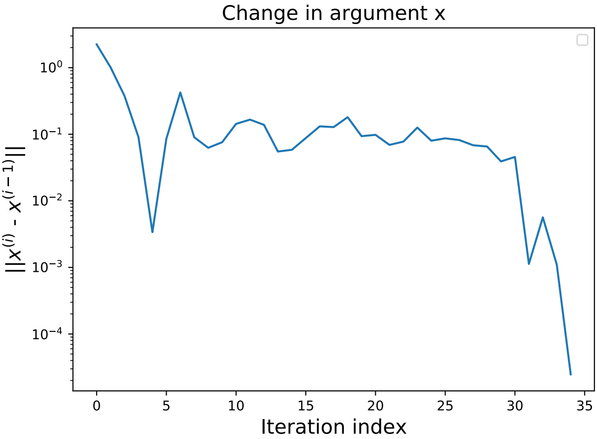
$$\min_{\mathbf{x}} f(\mathbf{x}) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2$$

The true minimum is $\mathbf{x} = (1, 1)$

Convergence plot



Convergence plot



Further reading

- More tutorials can be found at ▸ Link
- Tutorials to plot the optimization path ▸ Link