CS 519 Project Proposal: Visualization of Optimization Algorithms

netid: mmorais2

Proposal

Optimization algorithms seek to find the best solution x* from a set S such that $f(x^*) \leq f(x)$ for all x in S. For this project I will implement a handful of optimization algorithms, evaluate their performance on some well known test functions, and create visualizations of these algorithms in order to provide some insight into their relative strengths and weaknesses.

Algorithms

Algorithms will be selected to satisfy the following criteria:

- Representative of a general approach eg first-order methods vs. stochastic methods.
- Different in some way from the other methods which are chosen.
- Reasonable to implement in python with numpy, scipy, and autograd.
- Interesting to visualize.

My algorithm candidates are listed in the table below. I may reduce the number or choice depending on the quality of the visualization results obtained for each.

Method	Approach	Notes
Gradient Descent	First-Order Methods	
BFGS	Second-Order Methods	Approximated Hessian.
Simulated Annealing	Stochastic Methods	
Particle Swarm	Population Method	

Evaluation

In a literature review of test functions used to validate optimization algorithms Jamil and Yang (2013) outline the characteristics of test functions based on the following criteria:

- Modality: The number of ambiguous peaks in the landscape.
- Basins: A large area of little change surrounded by steep slopes.
- Valley: A narrow area of little change surrounded by steep slopes.
- Separability: Degree of independence of parameters in the search space.
- Dimensionality: Number of parameters in the search space.

My test function candidates are listed in the table below. I may reduce the number or choice depending on the quality of the visualization results obtained for each.

Name	Criteria Discussion	Notes
Rosenbrock	Continuous, **Differentiable**, Non-Separable, Scalable, **Unimodal**	Global minimum located in a valley. (wikipedia)
Stepint	Discontinuous, **Non-Differentiable**, Separable, Scalable, **Unimodal**	Stair-step landscape.
Bartels Conn	Continuous, **Non-Differentiable**, Non-Separable, Non-Scalable, **Multimodal**	
Goldstein Price or Himmelblau	Continuous, **Differentiable**, Non-separable, Non-Scalable, **Multimodal**	Several local minima. Large basin. (wikipedia: Goldstein Price) (wikipedia: Himmelblau)

Visualizations

Test Functions

I will create the following visualizations for each test function:

- 2D contour plot.
- 3D surface plot.

The plots should:

- Highlight local and global minima.
- Use perceptually uniform colormaps.

I plan to use the pyvista library (link) for both kinds of visualizations.

Algorithms

I will create the following visualizations for each algorithm solution:

- Animated algorithm solution superimposed on 2D contour plot.
- Animated algorithm solution superimposed on 3D surface plot.

The animations should demonstrate:
- Progress as a function of iteration number.
- Rate of convergence to solution.

Milestones

Name	Description	Weight (%)
Implement algorithms.	To maximize the learning, I prefer to use my own implementations of each algorithm, but I might resort to using a library if the results on the test functions require a more sophisticated implementation. I already have implementations of gradient descent (using autograd), BFGS, and simulated annealing, but particle swarm will be new.	20
Visualize Test Functions	• 2D contour plot. • 3D surface plot.	40
Visualize Algorithms	• Animated algorithm solution superimposed on 2D contour plot. • Animated algorithm solution superimposed on 3D surface plot.	40

References

Momin Jamil and Xin-She Yang, A literature survey of benchmark functions for global optimization problems, Int. Journal of Mathematical Modelling and Numerical Optimisation, Vol. 4, No. 2, pp. 150–194 (2013). DOI: 10.1504/IJMMNO.2013.055204

Mykel J. Kochenderfer and Tim A. Wheeler. 2019. Algorithms for Optimization. The MIT Press.

Test functions for optimization. Wikipedia.
https://en.wikipedia.org/wiki/Test_functions_for_optimization

Group

I will work on this project alone.