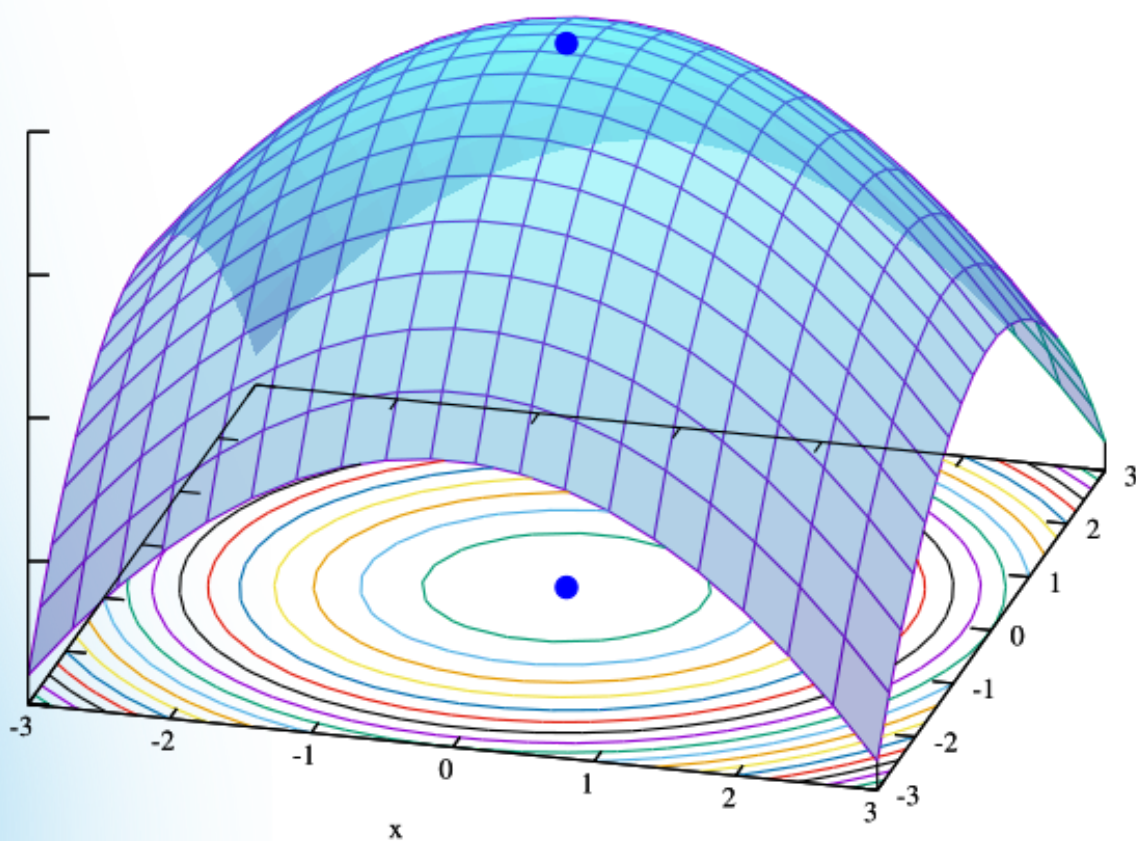


TOP 10 PYTHON OPTIMIZATION LIBRARIES in 2024



TIMUR BISMUKHAMETOV

1 **scipy.optimize** (★ 13.1k*) Project Link: [Link](#)

SciPy.optimize includes algorithms for constrained and unconstrained optimization, such as linear programming, nonlinear optimization, and root finding.

It's mature and widely used for general numerical computations.

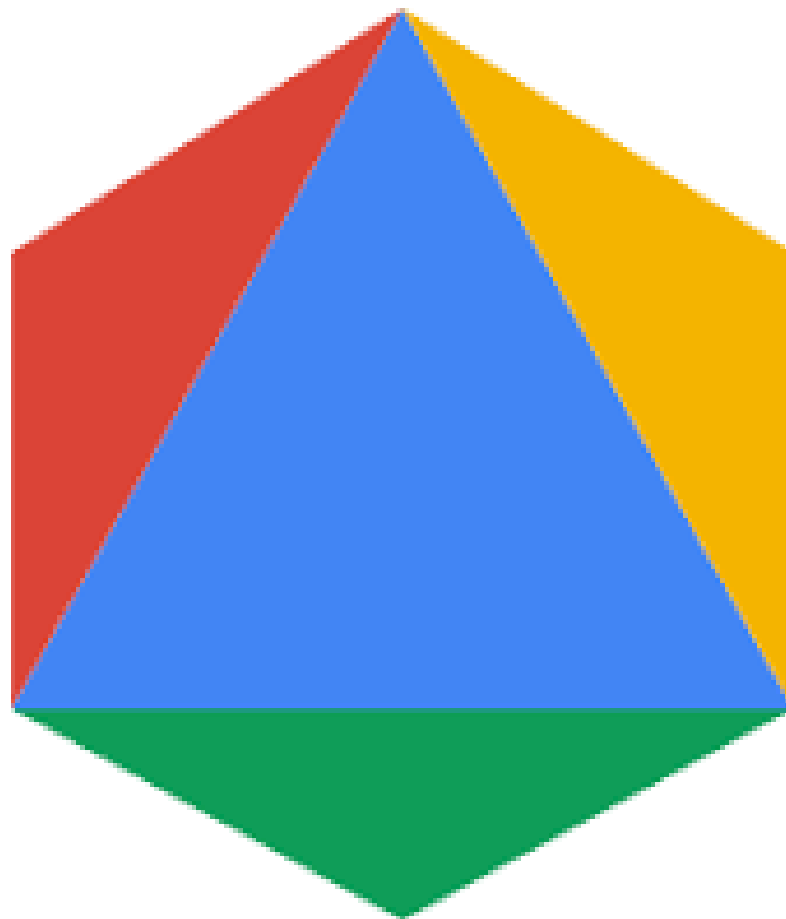


*Stars are recorded based on the entire SciPy library

2 Google OR-Tools (★11.2k) GitHub Link: [Link](#)

Google OR-Tools is designed for combinatorial optimization problems, including vehicle routing, scheduling, and resource allocation.

It supports linear programming, integer programming, and constraint programming, making it popular in operational research.



3 Optuna (★ 10.7) GitHub Link: [Link](#)

Optuna is an automatic optimization framework uses efficient search algorithms like Bayesian optimization and pruning techniques to find optimal solutions for a wide range of problems.

Beyond hyperparameter tuning, it can be used for complex multi-objective and constrained optimization tasks, making it suitable for applications in operational research, resource allocation, and scientific experimentation where optimal configurations are critical.



O P T U N A

4 HyperOpt (★7.2k) Project Link: [LINK](#)

Hyperopt is an optimization library that leverages algorithms like Random Search and Tree-structured Parzen Estimators (TPE) for efficient exploration of large search spaces.

Although commonly used for hyperparameter tuning, it's equally effective for broader optimization tasks, including resource scheduling, parameter selection in engineering, and other applications where gradient-free optimization is advantageous.

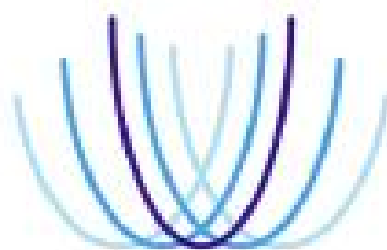


HYPEROPT

5 CVXPY (★ 5.4k) GitHub: [LINK](#)

CVXPY is a modeling library for convex optimization that lets users define and solve problems using simple syntax.

It supports a range of convex solvers and is used in fields like machine learning, economics, and operational research.

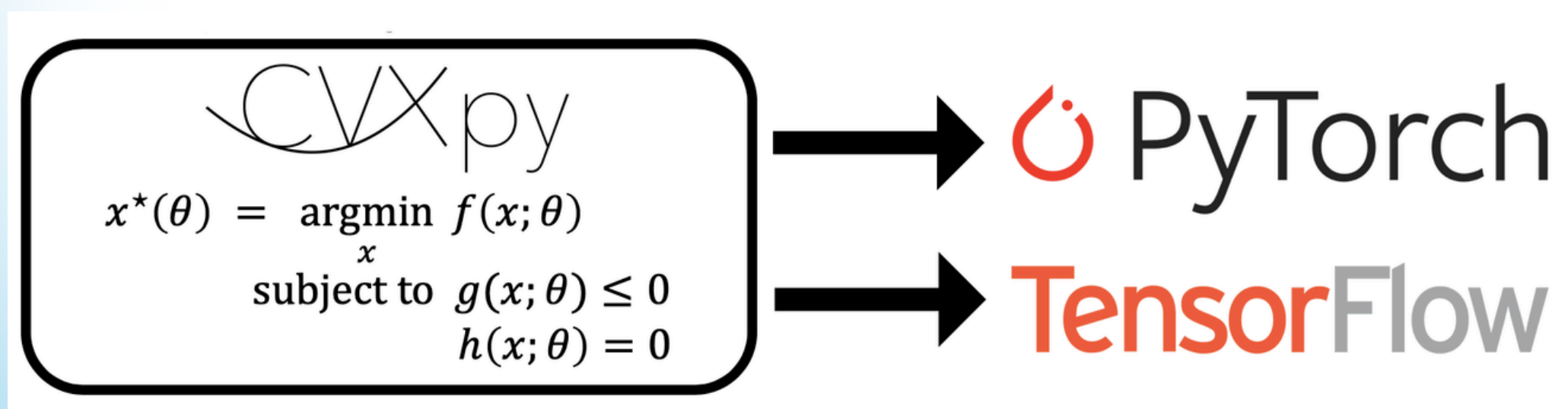


CVXPY

6 CVXPY Layers (★1.8k) GitHub: [LINK](#)

CVXPY Layers is an extension of CVXPY that allows for embedding optimization layers within deep learning models in frameworks like PyTorch and TensorFlow, enabling gradient-based optimization within neural networks.

This makes it particularly useful for tasks that require solving an optimization problem as part of the model's forward pass, such as structured prediction or control tasks.



PuLP(★ 2.1k) GitHub: [LINK](#)

PuLP is a Python library specifically for linear and integer programming.

PuLP is user-friendly and integrates with several solvers, making it ideal for prototyping and educational purposes.

It's lightweight yet effective for creating and solving small to medium-sized LP problems, widely used in resource allocation and scheduling.



8 Pyomo (★ 2k) GitHub: [LINK](#)

Pyomo is a flexible library for defining and solving mathematical optimization problems, covering linear, nonlinear, and mixed-integer programming.

It's widely used for large-scale optimization modeling in both academia and industry.

With its extensive support for complex constraints and multi-stage problems, Pyomo is ideal for applications in energy systems, supply chain optimization, and logistics planning.

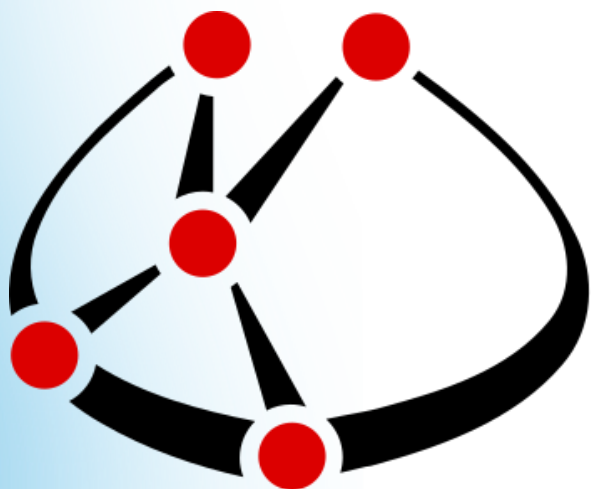


9 CasADi (★ 1.7k) GitHub: [LINK](#)

CasADi is a symbolic framework for automatic differentiation and optimization, especially suited for dynamic optimization and control applications.

It's commonly used in robotics and engineering for optimal control and real-time optimization.

CasADi's efficient handling of symbolic expressions and its compatibility with various solvers make it an excellent choice for solving complex, high-dimensional optimization problems in scientific computing and simulation.



CasADi

10 PyGAD (★ 1.7k) GitHub: [LINK](#)

PyGAD is a Python library for implementing Genetic Algorithms (GA) that is useful for solving optimization problems where traditional gradient-based approaches are insufficient, particularly for combinatorial or discrete optimization.

It supports a range of genetic operations like selection, crossover, and mutation, making it highly customizable for evolving solutions in applications such as feature selection.



P.S. Would you like to see practical tutorials with these libraries?



Comment “Want tutorial” under the post

I will make a tutorial if enough people are interested!



TIMUR BIKMUKHAMETOV

Want to see more lists like this?



Like post

Follow

Follow me

so I know the content resonates!



TIMUR BIKMUKHAMETOV