# Homework 1: Random Search

| Contact TA | Yiquan Xiao |
| --- | --- |
| Released | Tuesday, 9/12/2023 |
| Submission deadline | Midnight 11:59 PM, Thursday, 9/21/2023 |
| Total Marks | 100 |

## Task Descriptions

The first homework assignment has the following three programming tasks.

### Task 1: Implement random search algorithm

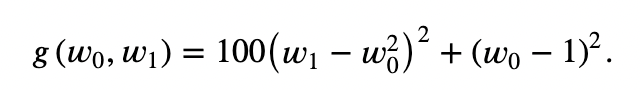
In the [hw1.py](https://drive.google.com/file/d/1TuNBJEWLNXXXOZzhszqnQIGSyOqaryx1/view?usp=sharing) file provided, you will find a function called: random\_search(...).

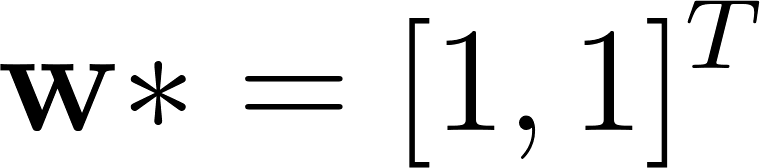
The function provides a skeleton for your production of the **random local search algorithm**.

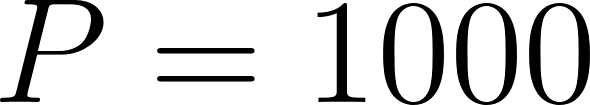
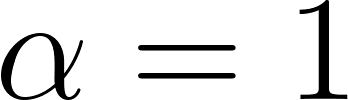
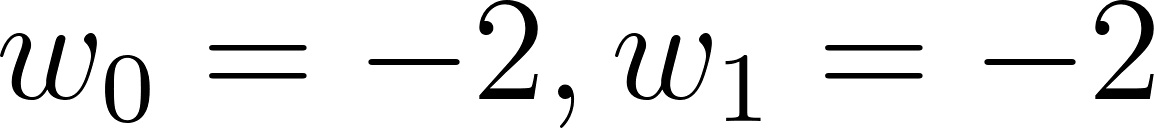
All parts marked "TODO" are for you to construct.

To run the file, you need to install Python and several other libraries. Please refer to the Appendix on how to configure the environment.

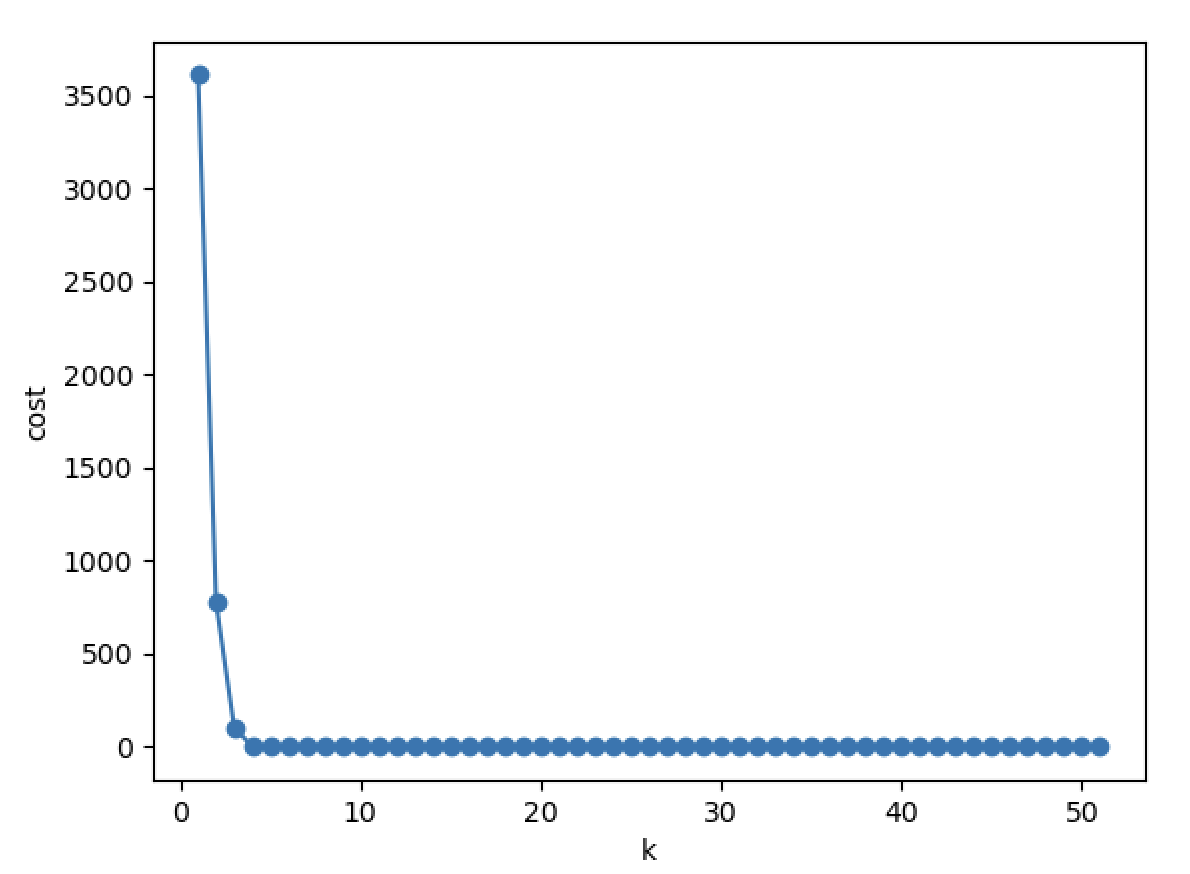
### Task 2: Optimize a function with random search

Use your completed random\_search function from the previous task to minimize the function:

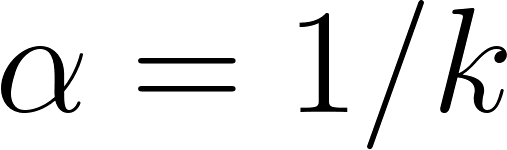
This function has a global minimum at the point [](https://www.codecogs.com/eqnedit.php?latex=%5Cmathbf%7Bw%7D*%3D%5B1%2C%201%5D%5ET#0) located in a very narrow and curved valley.

Run random search using [](https://www.codecogs.com/eqnedit.php?latex=P%3D1000#0), max iteration 50 steps, [](https://www.codecogs.com/eqnedit.php?latex=%5Calpha%20%3D%201#0), and initial parameters [](https://www.codecogs.com/eqnedit.php?latex=w_0%20%3D%20-2%2C%20w_1%20%3D%20-2#0).

Plot the cost history. The figure should be similar to this:



### Task 3: Optimize the same function with random search using diminishing steplength.

Re-run task 2 but with [](https://www.codecogs.com/eqnedit.php?latex=%5Calpha%20%3D%201%2Fk#0), where k is the iteration count.

This requires you to modify the “random\_search()” function to support diminishing steplength.

Compare the cost function history plots with random search with fixed steplength by plotting the cost function history plots for both runs in the same Figure.

Also report the final cost from both methods (with fixed learning rate and with diminishing learning rate)

## Deliverable

* **The completed source code hw1.py**
* **A pdf report**

The source code should be able to run by executing the command: python hw1.py

The PDF report should include:

* Result (both *weight\_history* and *cost\_history*) of running Task 1
* Cost history plot by running Task 2
* Cost history plots that compares random search with fixed steplength and random search with diminishing steplength. (Task 3) **Note: you should give us TWO cost history plots in order to get full credits for Task 3.** One is generated by using all data, and the other is generated by using the data starting from the 10th data. i.e. You should use *compare\_cost\_history(costs\_fixed[10:], costs\_diminished[10:])* to generate a plot of using data from the 10th data point. And then use *compare\_cost\_history(costs\_fixed, costs\_diminished)* to generate the other. You should include both of them in your report for Task 3!!!
* Final cost using a fixed steplength (Task 3)
* Final cost using a diminishing steplength (Task 3)

## Appendix: Environment Configuration Instructions

**1. What is Anaconda/Miniconda**

We strongly recommend you to use Anaconda or Miniconda. Before installing, you may want to know what Anaconda and Miniconda are?

**Anaconda:**

**What is it?** Anaconda is a free, open-source distribution of Python (and also R) for scientific computing and data science.

**Why use it?** Anaconda makes it easy to install many scientific libraries and tools (like Jupyter, TensorFlow, and Scikit-learn) with a single installation. It also offers a tool called 'conda' for package management and environment management, which helps you maintain isolated coding environments for different projects.

**Who is it for?** Beginners who want an all-in-one installation for data science tools and libraries in Python.

**Miniconda:**

**What is it?** Miniconda is a lightweight version of Anaconda. It only comes with the Python language and the 'conda' package manager, without the additional libraries and tools.

**Why use it?** Miniconda offers flexibility. You start with a minimal setup and only install what you need. This makes it lighter and faster than the full Anaconda distribution.

**Who is it for?** Those who want a minimalistic setup and prefer to handpick their libraries and tools.

**2. How to install Anaconda/Miniconda**

You can easily install Anaconda or Miniconda by just following the steps from [here](https://docs.anaconda.com/free/anaconda/install/index.html) or [here](https://docs.conda.io/projects/miniconda/en/latest/miniconda-install.html).

Quick links:

1. Installing [Anaconda on Windows](https://docs.anaconda.com/free/anaconda/install/windows/)
2. Installing [Anaconda on macOS](https://docs.anaconda.com/free/anaconda/install/mac-os/)
3. Installing [Anaconda on Linux](https://docs.anaconda.com/free/anaconda/install/linux/)

If you cannot successfully install Anaconda or Miniconda, please come to our office hours or post your problems on Campuswire.

**3. How to use Anaconda**

Anaconda has a graphical user interface (GUI) called the Navigator. So if you install Anaconda, you can either use Anaconda Navigator or command line interface (CLI). If you install Miniconda, you only have CLI.

**Should I use Anaconda Navigator or conda?**

Anaconda Navigator is a desktop application that is included with every installation of Anaconda Distribution. It is built on top of conda, the open-source package and environment manager, and allows you to manage your packages and environments from a graphical user interface (GUI). This is especially helpful when you’re not comfortable with the command line.

A command line interface (or CLI) is a program on your computer that processes text commands to do various tasks. Conda is a CLI program, which means it can only be used via the command line. On Windows computers, Anaconda recommends that you use the Anaconda Prompt CLI to work with conda. MacOS and Linux users can use their built-in command line applications.

**An introduction to Navigator and the command line**

Navigator and the CLI interact with conda in similar but distinct ways, and each have their benefits and drawbacks. Anaconda recommends that you learn the basics of both to determine what is preferable for your programming workflow. See [My first Python program: Hello, Anaconda!](https://docs.anaconda.com/free/anaconda/getting-started/hello-world/) to go through a short programming exercise and get a better idea for what you prefer.

**Getting started with Anaconda**

Getting started with Anaconda Distribution.

Quick links:

1. Getting started with [Navigator](https://docs.anaconda.com/free/navigator/getting-started/)
2. Getting started with [conda](https://conda.io/projects/conda/en/latest/user-guide/getting-started.html)

Some [tutorials](https://docs.anaconda.com/free/navigator/tutorials/) provided by Anaconda.

**Using Anaconda/Miniconda with IDEs**

Anaconda provides guidelines on how to use it in many integrated development environments (IDEs). From [here](https://docs.anaconda.com/free/anaconda/ide-tutorials/), select the IDE you want to use and review the corresponding guide.

Some other guide provided by IDEs:

1. Configure a conda virtual environment in [PyCharm](https://www.jetbrains.com/help/pycharm/conda-support-creating-conda-virtual-environment.html)
2. Using Python Environments in [Visual Studio Code](https://code.visualstudio.com/docs/python/environments)

**4. Environment Configuration**

For all assignments in this course, you should use Python of version at least 3.6. The code can run on my python version 3.9.

Third-party libraries allowed for this assignment:

1. [Numpy](https://numpy.org/doc/stable/)
2. [Matplotlib](https://matplotlib.org/)

You **CAN** use any of the [Python built-in libraries](https://docs.python.org/3/library/index.html), but you cannot **CANNOT** use third-party libraries other than those mentioned above for **this** assignment. If you have any questions about using a specific library, please stop by our office hours or ask your question on Campuswire.