

# **Практическая работа №8 от 15.05.2020**

## **Эволюционная оптимизация**

### ***Цель работы***

Практическое использование метода дифференциальной эволюции.

### ***Задачи работы***

1. Изучить теоретические основы метода дифференциальной эволюции.
2. Научиться использовать метод дифференциальной эволюции.
3. Научиться оценивать эффективность метода дифференциальной эволюции.

### ***Перечень обеспечивающих средств***

1. ПК.
2. Учебно-методическая литература.
3. Задания для самостоятельного выполнения.

### ***Общие теоретические сведения***

#### **Подходы и методы**

Базовый генетический алгоритм на данный момент применяется редко из-за его ограничений. Однако сам подход эволюционной оптимизации используется широко за счёт применения модификаций генетического алгоритма.

В данной работе мы рассмотрим метод дифференциальной эволюции, реализованный в пакете `scipy` языка Python.

Стоит отметить, что существует большое количество библиотек так или иначе реализующих генетический алгоритм либо его модификации. Также, благодаря простоте генетического алгоритма, легко реализовать его самостоятельно.

#### **Решаемая задача**

Метод дифференциальной эволюции позволяет найти глобальный минимум функции многих переменных на некоторой области определения (т. е. на множестве значений переменных).

Для работы метода достаточно, чтобы можно было вычислить значение функции в любой точке области определения. Существования производной, непрерывности и т. п. оптимизируемой функции не требуется.

#### **Алгоритм работы метода дифференциальной эволюции**

Алгоритм работы метода (в реализации `scipy`) следующий:

1. Задаем оптимизируемую функцию  $f(X)$  и ее область значений.
2. Формируем первое поколение векторов из области значений. Это можно

сделать несколькими способами:

- а) «latinhypercube» – равномерное покрытие всей области определения,
  - б) «random» – случайная выборка,
  - с) явно задать координаты векторов.
3. Для каждого вектора в текущем поколении создается «тестовый» вектор. Есть несколько стратегий построения «тестовых» векторов, каждая лучше подходит для определенного типа задач. Вариант по-умолчанию – «best1bin».
  4. Сравнивается значение оптимизируемой функции для текущего и «тестового» векторов, в следующую популяцию попадает тот, для которого значение функции меньше. Таким образом размер популяции всегда остается постоянным.
  5. Если не достигнут критерий остановки, то переходим к шагу №3. Критерием остановки может быть количество поколений, изменение ошибки по сравнению с предыдущим поколением и т. д.

Основные настроечные параметры метода:

- strategy – эволюционная стратегия, задающая правила формирования новой популяции на основе существующей,
- maxiter – максимальное количество поколений,
- popsize – размер популяции,
- init – способ задания первого поколения,
- mutation – если задано как число из интервала  $[0,2]$ , то используется как коэффициент случайных изменений в векторах; если задан кортеж (min, max), то используется режим сглаживания, при котором коэффициент меняется от min до max с ростом номера поколения,
- recombination – число в интервале  $[0,1]$ , которое задает вероятность рекомбинации при формировании нового поколения.

Каждый параметр балансирует между двумя крайностями:

- большой радиус поиска, хорошие шансы найти глобальный минимум, но долгая сходимость,
- маленький радиус поиска, большая вероятность остановиться в локальном минимуме, но быстрая сходимость.

## Задание

- Сделайте форк репозитория <https://github.com/mosalov/DifferentialEvolution>.
- Откройте репозиторий в Binder(<https://mybinder.org/>) и запустите файл «differential\_evolution.ipynb».
- Выполните представленный в файле код (кроме последнего блока), чтобы понять, как правильно пользоваться методом differential\_evolution библиотеки scipy на примере функции Бута:  
$$f(x,y) = (x + 2y - 7)^2 + (2x + y - 5)^2$$
- Измените оптимизируемую функцию на функцию Химмельблау (ее код уже есть в файле):  
$$f(x,y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$$
- Выполните последний блок кода (поиск всех четырех минимумов) для различных значений параметра mutation. Как минимум используйте значения 0,5; 1 и 1,5. Проинтерпретируйте (объясните) полученный результат.
- Реализуйте функцию Матьяса:  
$$f(x,y) = 0,26(x^2 + y^2) - 0,48xy$$
- Установите параметры алгоритма popsize = 10 и init = 'random'.
- Оцените скорость работы алгоритма (по количеству необходимых поколений) в зависимости от значения параметра recombination. Как минимум используйте значения 0,25; 0,5; 0,75. Проинтерпретируйте (объясните) полученный результат.
- Сохраните файл Jupyter notebook с названием «Задание 8.ipynb» и загрузите его в репозиторий.

## Требования к предоставлению результатов

Требуется представить результаты в виде письма на адрес [mosalov.op@ut-mo.ru](mailto:mosalov.op@ut-mo.ru) с указанием ФИО и ссылки на репозиторий с сохраненным файлом Jupyter notebook.

## Литература

1. [https://ru.wikipedia.org/wiki/Дифференциальная\\_эволюция](https://ru.wikipedia.org/wiki/Дифференциальная_эволюция)
2. [https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential\\_evolution.html](https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential_evolution.html)
3. [https://ru.wikipedia.org/wiki/Тестовые\\_функции\\_для\\_оптимизации](https://ru.wikipedia.org/wiki/Тестовые_функции_для_оптимизации)