Отчет по лабораторной работе №4

Работа выполнили:

Химченко Максим группа М3232

Товмасян Арман группа М3232

Кистер Артемий группа М3232

Постановка задачи:

1) Разберите теоретическое описание и реализуйте метод стохастической оптимизации (к примеру, метод имитации отжига);
2) Сравните его эффективность на методах и примерах из лаб. 1 и/или лаб 2. Приведите примеры, иллюстрирующие разницу в результатах и эффективности (количество вызовов функции, скорости) применения методов из лаб. 1 и/или лаб 2 и методов стохастической оптимизации.

Введение

В данном отчете рассматриваются методы стохастической оптимизации и инструменты библиотеки Optuna для оптимизации функций. Основное внимание уделяется методу имитации отжига и его сравнению с другими методами оптимизации, такими как градиентный спуск и метод Ньютона. Применение этих методов будет проиллюстрировано на примере функции Розенброка.

Основное задание

Теоретическое описание метода имитации отжига

Метод имитации отжига - стохастический метод оптимизации, который основывается на аналогии с процессом отжига металлов. Этот метод эффективен для поиска глобального минимума функции в многомерных пространствах, особенно когда функция имеет множество локальных минимумов.

Основные этапы алгоритма:

1) Инициализация: Выбор начального состояния и температуры.
2) Генерация нового состояния: Случайное изменение текущего состояния.
3) Оценка: Вычисление значения целевой функции для нового состояния.
4) Прием решения: Решение о принятии нового состояния зависит от разности значений функции и текущей температуры.
5) Снижение температуры: Плавное уменьшение температуры по определенному закону.
6) Повторение: Возврат к шагу 2 до выполнения условия остановки.

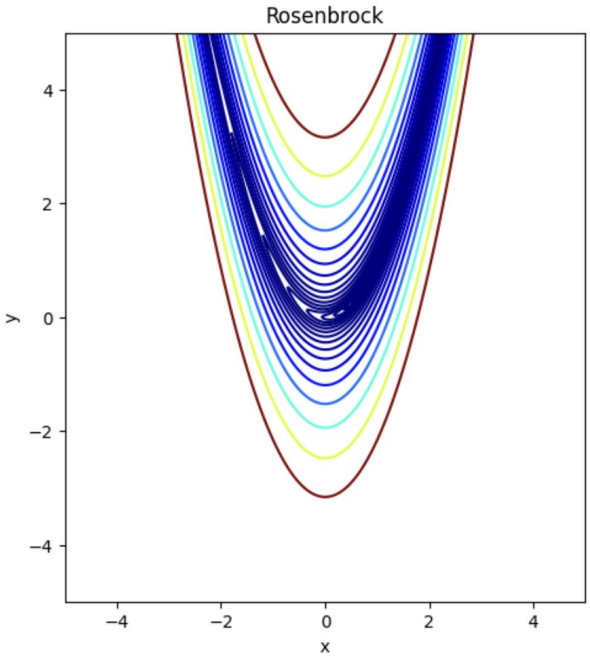
Применение метода имитации отжига на функции Розенброка:

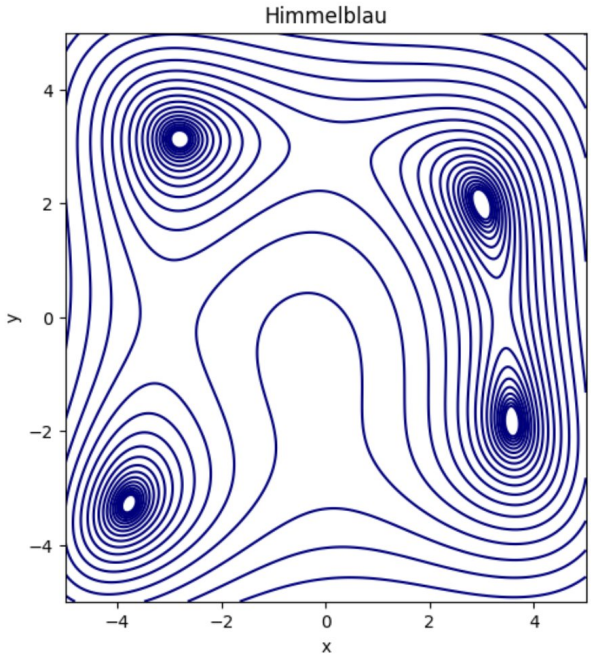
Функция Розенброка используется для тестирования методов оптимизации благодаря наличию узкого, изогнутого канала, ведущего к глобальному минимуму. Ее математическое представление:

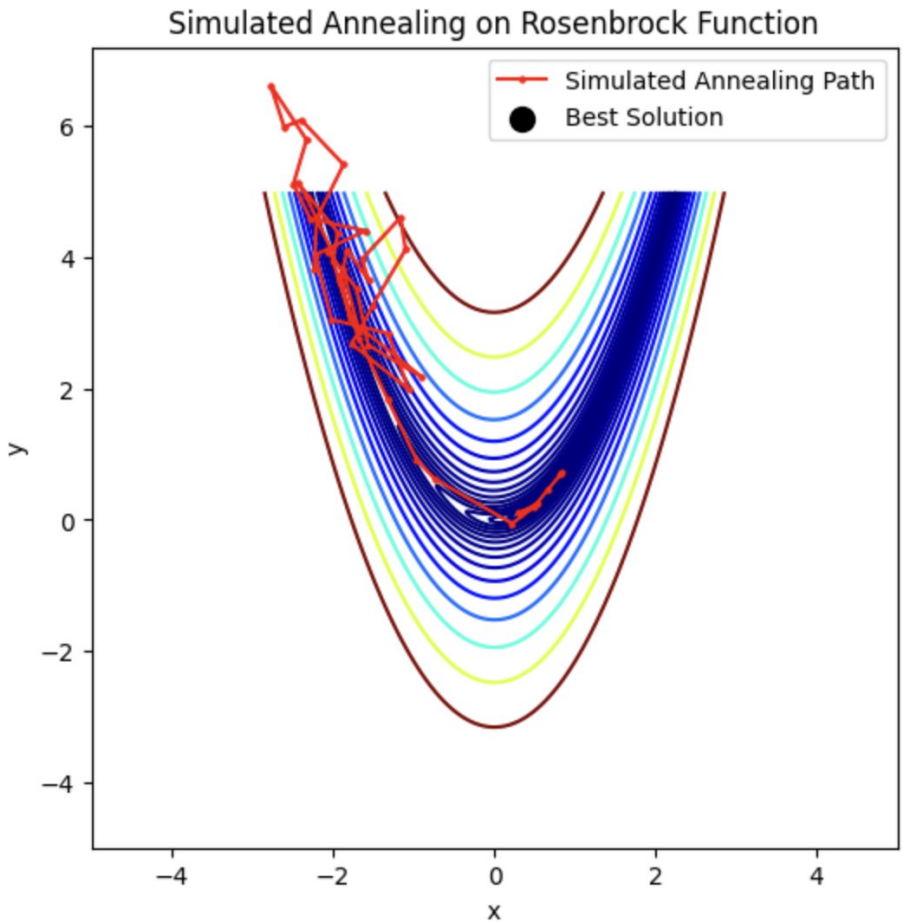
$f(x,y)$=(1−x)^2+100(y−x2)^2

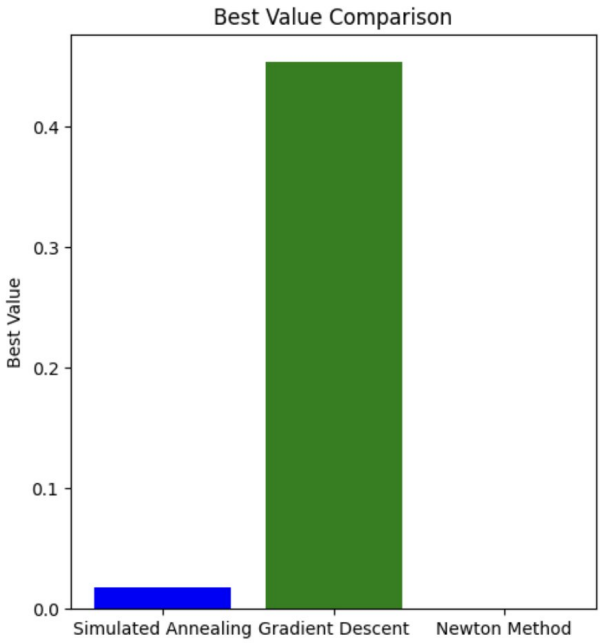
Сравнение с другими методами

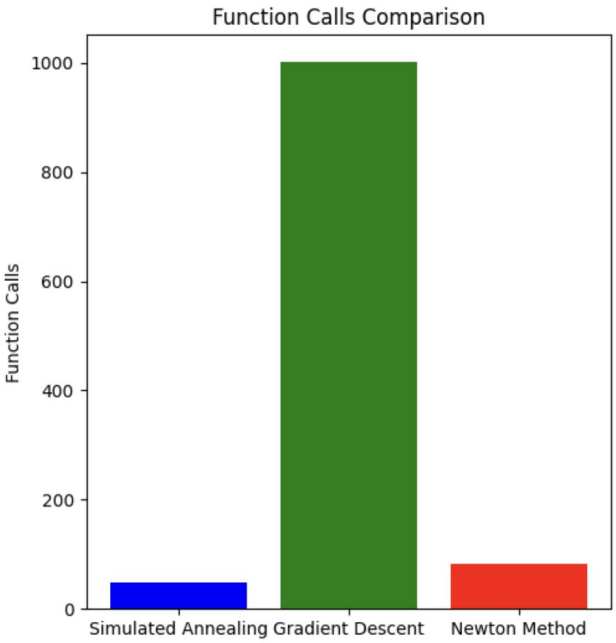
Для сравнения используем градиентный спуск и метод Ньютона на той же функции.











	Method	Best Value	Function Calls
0	Simulated Annealing	1.781138e-02	49
1	Gradient Descent	4.535290e-01	1001
2	Newton Method	3.031438e-10	83

Выводы из сравнения

1) Simulated Annealing:
   - Находит приемлемое решение с наименьшим количеством вызовов функции.
   a. Находит приемлемое решение с наименьшим количеством вызовов функции.
   b. Хорошо работает на функциях с множеством локальных минимумов
   c. Преимущество: Стохастический характер позволяет избежать локальных минимумов
   d. Недостаток: Медленная сходимость к точному решению по сравнению с детерминированными методами
2) Градиентный спуск
   a. Требует наибольшее количество вызовов функции
   b. Подвержен застреванию в локальных минимумах
   c. Преимущество: Простота реализации и понимания

Общая информация к дополнительным заданиям

Optuna – это библиотека для автоматизированной настройки гиперпараметров, предназначенная для эффективного поиска оптимальных параметров моделей машинного обучения. Основные особенности Optuna включают легкость использования, мощные возможности визуализации и поддержку различных алгоритмов оптимизации.

Основные методы и концепции, использованные в примере

1. **Создание исследования (`create_study`)**:

   - study = optuna.create_study(direction='minimize')

   - Создает объект исследования (`study`), который определяет направление оптимизации (минимизация или максимизация).

2. **Определение целевой функции (`objective`)**:

   - Целевая функция (`objective`) используется для оценки производительности комбинаций гиперпараметров.

   - Внутри целевой функции используются методы для выбора гиперпараметров, такие как trial.suggest_uniform для равномерного распределения или trial.suggest_loguniform для логарифмического распределения.

3. **Оптимизация (`optimize`)**:

   - study.optimize(objective, n_trials=100)

   - Запускает процесс оптимизации, выполняя заданное количество испытаний (trials), где каждая итерация оценивает новые комбинации гиперпараметров.

4. **Получение результатов**:

   - study.best_trial возвращает наилучший найденный результат, включая значение функции и параметры, которые к нему привели.

Краткие шаги

1. **Определение целевой функции (`objective`)**:

   - Определите функцию, которую нужно минимизировать или максимизировать.

   - Используйте методы trial.suggest_* для выбора гиперпараметров в пределах заданных диапазонов.

2. **Создание исследования (`create_study`)**:

   - Создайте объект исследования с указанием направления оптимизации (`minimize` или `maximize`).

3. **Запуск оптимизации (`optimize`)**:

   - Запустите оптимизацию, задав количество испытаний (trials).

4. **Анализ результатов**:

   - Извлеките лучшие результаты и параметры, используя study.best_trial.

Optuna предоставляет удобный и мощный инструмент для автоматизации настройки гиперпараметров, что позволяет улучшить производительность моделей машинного обучения и других оптимизационных задач.

Дополнительное задание 1:

Параметры вызываемых библиотечных функций

optuna.create_study():

- direction: Направление оптимизации ('minimize' или 'maximize'). В данном случае используется 'minimize', так как цель состоит в минимизации значения функции.

study.optimize():

- objective: Целевая функция для оптимизации. Она принимает объект trial, настраивает гиперпараметры и возвращает значение целевой функции.
- n_trials: Количество попыток оптимизации. В данном случае использовано 100 попыток.

Анализ результатов

Лучший результат: Значение функции -3.903161772999548 при x=−0.30676487955268944. Это значение является глобальным минимумом для данной функции.

Эффективность Optuna: Optuna эффективно нашла оптимальное значение за 100 попыток, что свидетельствует о высокой производительности библиотеки при поиске глобальных минимумов в сложных функциях.

Использование метода имитации отжига: В предыдущих заданиях метод имитации отжига также показал хорошие результаты, но требовал большего количества вызовов функции по сравнению с Optuna.

Дополнительное задание 2:

Параметры вызываемых библиотечных функций

optuna.create_study():

- direction: Направление оптимизации ('minimize' или 'maximize'). В данном случае используется 'minimize', так как цель состоит в минимизации значения функции Розенброка.

study.optimize():

- objective: Целевая функция для оптимизации. Она принимает объект trial, настраивает гиперпараметры и возвращает значение целевой функции.
- n_trials: Количество попыток оптимизации. В данном случае использовано 100 попыток.

optuna.create_study():

- direction: Направление оптимизации ('minimize' или 'maximize'). В данном случае используется 'minimize', так как цель состоит в минимизации значения функции.

study.optimize():

- objective: Целевая функция для оптимизации. Она принимает объект trial, настраивает гиперпараметры и возвращает значение целевой функции.
- n_trials: Количество попыток оптимизации. В данном случае использовано 100 попыток.

Анализ результатов

Лучший результат: Значение функции Розенброка 7.054912598812796e-06, что близко к глобальному минимуму.

Лучшие гиперпараметры: Скорость обучения 6.154740502689153e − 06 и максимальное количество итераций 5428. Это указывает на оптимальные параметры для достижения наилучшего результата в данном случае.