

СОГЛАСОВАНО

Сторона ЗАКАЗЧИКА

УТВЕРЖДАЮ

Сторона ИСПОЛНИТЕЛЯ

Профессор кафедры
ИАНИ ННГУ, д.т.н.

Д. В. Попов
« ____ » _____ 2022 г.

Н.В. Старостин
« ____ » _____ 2022 г.

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ РЕШЕНИЯ ЗАДАЧИ РЕДУКЦИИ
ПРОСТРАНСТВА МНОГОМЕРНЫХ ФУНКЦИЙ**

Руководство программиста

Этап 2. Разработка программной документации

**ОКР «РАЗРАБОТКА НЕЙРОННОЙ СЕТИ СПЕЦИАЛЬНОГО ВИДА
(АВТОЭНКODЕРА) ДЛЯ РЕШЕНИЯ ЗАДАЧИ РЕДУКЦИИ ПРОСТРАНСТВА
МНОГОМЕРНЫХ ФУНКЦИЙ»**

(Шифр ПО «Enc»)

Ответственный исполнитель

_____ В.А. Куликов

« ____ » _____ 2022 г.

2022 г.

АННОТАЦИЯ

Руководство программиста представляет собой информацию по содержанию, правилам работы и настройке программы «Enc». Программа выполняет запуски разного типа автоэнкодеров над различными функциями, подсчитывает ошибку и выводит результаты работы в выходные файлы.

Содержание

1. Общие сведения о программе.....	4
2. Характеристики программы.	5
3. Обращение к программе.....	6
4. Входные и выходные данные.	8

1. Общие сведения о программе.

1.1. Назначение программы.

ПО «Enc» предназначен для решения задачи редукции пространства многомерных функций.

1.2. Функции программы.

ПО «Enc» выполняет следующие операции:

- Считывание параметров выполнения программы из консоли.
- Считывание параметров и весов для автоэнкодера, при условии наличия файлов, содержащих эту информацию.
- Генерация случайных данных для параметров функции одним из трех методов.
- Обучение автоэнкодеров.
- Оптимизация параметров автоэнкодера методом полного перебора (не используется из-за долгого времени работы).
- Оптимизация параметров автоэнкодера алгоритмом эффективной глобальной оптимизации.
- Запуск автоэнкодеров на параметрах, полученных из входных файлов.
- Запись в файлы параметров и весов, полученных при обучении автоэнкодера.
- Сохранение диаграмм отклонений для каждого из параметров.

1.3. Системные требования.

Для функционирования ПО «Enc» ПЭВМ должны удовлетворять следующим требованиям: процессор Ryzen 5 или Intel i5, видеокарта GTX 1050Ti, оперативная память не менее 8 ГБ DDR4, HDD не менее 124 GB, клавиатура, мышь, интернет-доступ.

ПО «Enc» должно быть работоспособно при установленном python3, numpy, smt, tensorflow, sobol_seq, keras.

2. Характеристики программы.

ПО «Enc» является консольным приложением. Все возможные параметры для передачи в командную строку описаны в разделе «Обращение к программе». Основные характеристики ПО «Enc»:

1. Программа состоит из следующих модулей:
 - DataGenerator – Данный класс генерирует случайные данные для параметров функции тремя различными способами: полностью случайно в пределах области определения, методом sobol, методом латинского гиперкуба LSH.
 - Normalizer – модуль, реализующий класс нормировки/денормировки данных.
 - Function – модуль реализует класс функции, а также содержит стандартные функции func_1, func_2, func_3, func_4.
 - AutoEncoder_class – модуль содержит класс автоэнкодера и позволяет создать автоэнкодеры всех типов, а также содержит все необходимые функции для работы автоэнкодера.
 - Optimizer – модуль содержит класс оптимизации параметров автоэнкодера, который по результатам запусков автоэнкодеров, определяет лучшие параметры для выбранной функции.
 - Error_class – модуль реализует класс подсчета ошибки на основе средней абсолютной ошибки.
 - Calculate_error – модуль, предназначенный для запуска из консоли и вывода ошибок.
 - Training_model – модуль, предназначенный для запуска из консоли и обучения автоэнкодеров.
2. Сохранение параметров автоэнкодеров осуществляется в файлы .txt (Saved Models/Params). Сохранение весов автоэнкодеров осуществляется в файлы .h5 (Saved Models/Weights). Сохранение графиков ошибок для автоэнкодеров с различными параметрами осуществляется в файлы .png (Saved Models/Graphs).
3. Валидация результатов обучения оценивается с помощью ПМ Calculate_error.
4. Основные функции программы:
 - Обучение автоэнкодера для выбранной функции.

- Запуск автоэнкодера на готовых параметрах и весах, а также вывод ошибок.

3. Обращение к программе.

Высокоуровневый дизайн представленный на рисунке 1 имеет два сценария запуска:

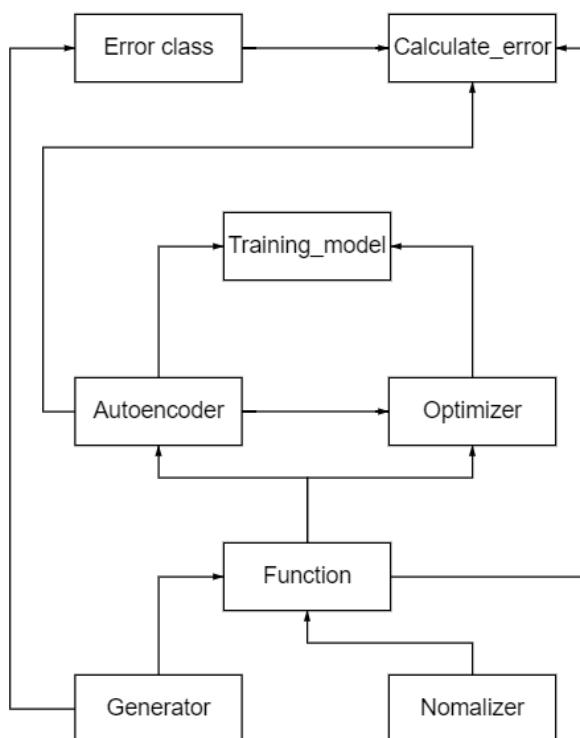


Рисунок 1 – Высокоуровневый дизайн.

Сценарий 1. Обучение автоэнкодера.

Для запуска программы необходимо:

- открыть командную строку ОС;
- перейти в директорию с исполняемым файлом программы командой:

```
>cd C:\Users\{ИМЯ_ПОЛЬЗОВАТЕЛЯ}\encoderProject\Code\Scripts
```

- прописать команду:

```
> python3.9 training_models.py [-h] [-f {func_1, func_2, func_3, func_4, all}] [-a {dense, deep, vae, all}] [-i [ITER], --iter [ITER]]
```

Сначала вызывается скрипт `training_models.py`, затем прописываются возможные аргументы:

«-h» – на консоль выводится все возможные аргументы

«-f» – выбор функции нейронной сети

- `func_1`
- `func_2`
- `func_3`
- `func_4`
- `all`

«-a» – выбор автоэнкодера для нейронной сети, где

- `dense` – сжимающий автоэнкодер
- `deep` – глубокий автоэнкодер
- `vae` – вариационный автоэнкодер
- `all` – использование всех автоэнкодеров для обучения

«-i» – Количество эпох подбора гиперпараметров автоэнкодера (по умолчанию 25)

Описание работы программных модулей:

1. Пользователь вводит:

```
> python3.9 training_models.py [-h] [-f {func_1, func_2, func_3, func_4, all}] [-a {dense, deep, vae, all}] [-i [ITER], --iter [ITER]]
```
2. В модуле `training_model` создается экземпляр тестируемой функции.
3. В модуле `training_model` создается экземпляр класса-оптимизатора.
4. В модуле `optimize_class` происходит генерация параметров и нормировка данных.
5. В модуле класса-оптимизатора создается экземпляр класса автоэнкодера с выбранными параметрами.
6. Класс-оптимизатор начинает подбор параметров.
7. Класс-оптимизатор производит обучение автоэнкодеров с выбранными параметрами и подает на выход наилучшую по отклонению комбинацию параметров автоэнкодера.
8. Класс-оптимизатор сохраняет веса и параметры автоэнкодеров в файлы.

Сценарий 2. Подсчет ошибки и вывод графиков.

1. Пользователь вводит:

- ```
> python3.9 calculate_error.py [-h] [-f {func_1, func_2, func_3, func_4, all}] [-a {dense, deep, vae, all}]
```
2. В модуле `calculate_error` создается экземпляр тестируемой функции.
  3. В модуле `calculate_error` создается экземпляр класса автоэнкодера с выбранными параметрами и загруженными весами.
  4. В модуле `calculate_error` создается экземпляр класса подсчета ошибки.
  5. С помощью модуля `error_class` производится подсчет средних общих ошибок и ошибок по параметрам.
  6. В модуле `error_class` сохраняется график отклонений по параметрам.

## 4. Входные и выходные данные.

Согласно сценариям обращения к программе, существуют два сценария, которые порождают входные/выходные данные.

### Сценарий 1.

Входные параметры:

- Выбранный тип функции
- Выбранный тип автоэнкодера
- Количество эпох.

Выходные параметры:

Файлы с весами обученного автоэнкодера представлены на рисунке 2.

- Веса обученного автоэнкодера.






| Имя                                                                                                         | Дата изменения   | Тип                | Размер |
|-------------------------------------------------------------------------------------------------------------|------------------|--------------------|--------|
|  func_1_ego_dense_8_4.h5 | 16.01.2022 19:06 | Файл "H5"          | 15 КБ  |
|  func_1_ego_dense_8_5.h5 | 16.01.2022 19:06 | Файл "H5"          | 15 КБ  |
|  func_1_ego_dense_8_6.h5 | 16.01.2022 19:05 | Файл "H5"          | 15 КБ  |
|  func_1_ego_dense_8_7.h5 | 14.01.2022 18:55 | Файл "H5"          | 15 КБ  |
|  readme.txt              | 14.01.2022 14:58 | Текстовый докум... | 1 КБ   |

Рисунок 2. - Файлы с весами обученного автоэнкодера.

- Оптимальные гиперпараметры автоэнкодера по итогам процесса оптимизации параметров показаны на рисунке 3.



```

func_1_ego_dense_8_5.txt — Блокнот
Файл Правка Формат Вид Справка
func name: func_1
epochs: 40
batch: 59
encoded dim: 5
sample split: 0.8286085227820608

```

Рисунок 3 – Оптимальные гиперпараметры автоэнкодера

### Сценарий 2.

Входные параметры:

- Веса обученного автоэнкодера.
- Оптимальные параметры автоэнкодера по итогам процесса оптимизации параметров.
- Выбранный тип функции.
- Выбранный тип автоэнкодера.

Выходные параметры:

- Графики отклонений по параметрам показаны на рисунке 4.

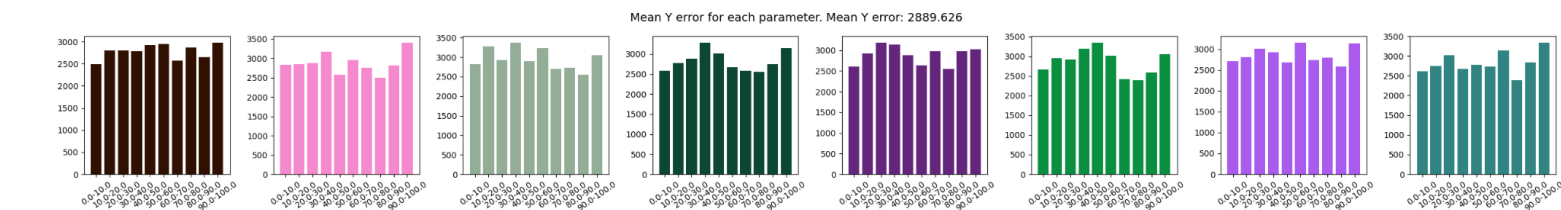


Рисунок 4 – Графики отклонений по параметрам