|  |  |
| --- | --- |
| СОГЛАСОВАНО | УТВЕРЖДАЮ |
| Сторона ЗАКАЗЧИКА  Попов Д.В.  « » 2021 г. | Сторона ИСПОЛНИТЕЛЯ  Профессор кафедры  ИАНИ ННГУ, д.т.н.  Н.В. Старостин  « » 2021 г. |

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

**Руководство оператора**

**Этап 2.3 Разработка программной документации**

**«Разработка нейронной сети специального вида (автоэнкодера) для решения задачи редукции пространства многомерных функций»**

**(Шифр ПО «Enc»)**

Ответственный исполнитель

\_\_\_\_\_\_\_\_\_\_\_\_ В.А. Куликов

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

2022 г.

АННОТАЦИЯ

В данном руководстве описана структура, принципы работы, базовые понятия и интерфейс программного обеспечения «Enc», а также определены условия, необходимые для эффективного функционирования программного обеспечения и указана последовательность действий оператора при запуске и выполнении программы.

СОДЕРЖАНИЕ

[1. НАЗНАЧЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 4](#_Toc72342600)

[2. ТРЕБОВАНИЯ К ПРОГРАММНОМУ И АППАРАТНОМУ ОБЕСПЕЧЕНИЮ 4](#_Toc72342601)

[2.1 Минимальный состав аппаратных средств 4](#_Toc72342602)

[2.2 Минимальный состав программных средств 4](#_Toc72342603)

[3. ВЫПОЛНЕНИЕ ПРОГРАММЫ 4](#_Toc72342604)

[3.1 Загрузка и запуск программы 4](#_Toc72342605)

[3.2 Запуск программы отрисовки 5](#_Toc72342606)

[3.3 Этапы работы программы 6](#_Toc72342607)

[3.4 Проверка корректности исходных данных 8](#_Toc72342608)

[4. СООБЩЕНИЯ ОПЕРАТОРУ 8](#_Toc72342609)

# **НАЗНАЧЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

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

# **ТРЕБОВАНИЯ К ПРОГРАММНОМУ И АППАРАТНОМУ ОБЕСПЕЧЕНИЮ**

## Минимальный состав аппаратных средств

Для функционирования ПО «Enc» ПЭВМ должны удовлетворять следующим требованиям: процессор Ryzen 7, видеокарта RTX 2080Ti, оперативная память не менее 16 ГБ DDR4, SSD m2 не менее 1 GB, клавиатура, мышь, интернет-доступ.

## Минимальный состав программных средств

ПО «Enc» должно быть разработано с использованием языка программирования Python.

ПО «Enc» должно быть работоспособно под управлением следующих операционных систем: Windows 10.

ПО «Enc» должно быть работоспособно при установленном python3, numpy, smt, tensorflow, sobol\_seq (git+https://github.com/naught101/sobol\_seq@v0.2.0#egg=sobol\_seq – установить библиотеку при помощи команды !pip install), keras.

# **ВЫПОЛНЕНИЕ ПРОГРАММЫ**

## Загрузка и запуск программы тренировки нейронной сети

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:

> python training\_models.py [-h] [-f {func\_1, func\_2, func\_3, func\_4, all}] [-a {dense, deep, vae, all}]

где:  
сначала вызывается скрипт training\_models.py,   
затем прописываются возможные аргументы:

* -h – на консоль выводится все возможные аргументы
* -f – выбор функции нейронной сети
  + - func\_1, где , размерность пространства - 8, параметров ф-ции - 8
    - func\_2, где , размерность пространства 4, параметров ф-ции - 2 - спектр
    - func\_3, где , размерность пространства 6, параметров ф-ции - 3 - сдвинутый спектр
    - func\_4, где , размерность пространства 10, параметров ф-ции - 6 - сдвинутый спектр
    - all - использование всех функций для обучения
* -a – выбор автоэнкодера для нейронной сети, где
  + - dense – сжимающий автоэнкодер
    - deep – глубокий автоэнкодер
    - vae – вариационный автоэнкодер
    - all – использование всех автоэнкодеров для обучения

## Запуск нейронный сети для получения результатов

Для запуска программы необходимо:

* произвести тренировку нейронный сети для нужной функции и автоэнкодера
* открыть командную строку ОС;
* перейти в директорию с исполняемым файлом программы командой:

*>*cd C:\Users\*{ИМЯ\_ПОЛЬЗОВАТЕЛЯ}*\encoderProject\Code\Scripts

* прописать команду:

> python calculate\_error.py [-h] [-f {func\_1, func\_2, func\_3, func\_4, all}] [-a {dense, deep, vae, all}]

где:  
сначала вызывается скрипт training\_models.py,   
затем прописываются возможные аргументы:

* -h – на консоль выводится все возможные аргументы
* -f – выбор функции нейронной сети
  + - func\_1, где , размерность пространства - 8, параметров ф-ции - 8
    - func\_2, где , размерность пространства 4, параметров ф-ции - 2 - спектр
    - func\_3, где , размерность пространства 6, параметров ф-ции - 3 - сдвинутый спектр
    - func\_4, где , размерность пространства 10, параметров ф-ции - 6 - сдвинутый спектр
    - all - использование всех функций для получения результатов
* -a – выбор автоэнкодера для нейронной сети, где
  + - dense – сжимающий автоэнкодер
    - deep – глубокий автоэнкодер
    - vae – вариационный автоэнкодер
    - all – использование всех автоэнкодеров для получения результатов

## Этапы работы программы

В случае успешного завершения работы программы по тренировке нейронной сети консоль выведет сообщение с результатами тестирования нейронной сети (Рис. 1).

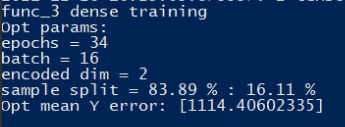


Рисунок 1.

После этого в папке encoderProject-master\Saved models\Params и encoderProject-master\Saved models\Weights будут сохранены веса для данной нейронной сети (рис. 2 и 3)

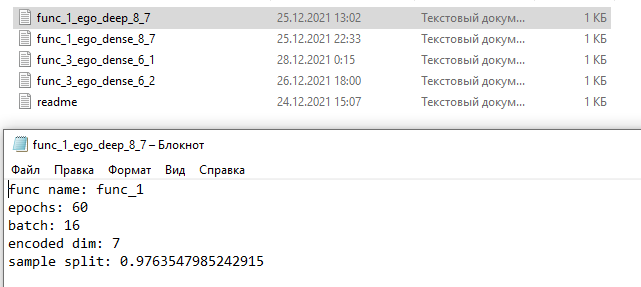


Рисунок 2.

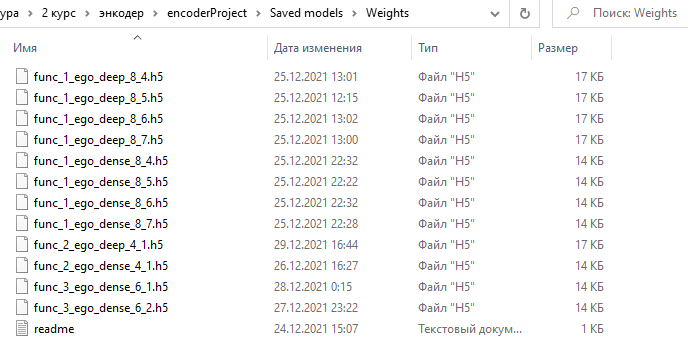


Рисунок 3.

При успешном запуске обученной сети на консоль выведется следующее сообщение:

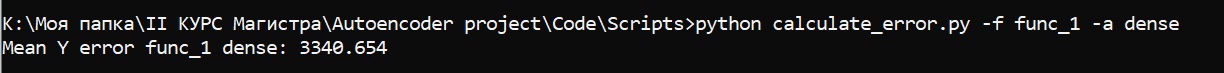


Рисунок 4.

Найти полученный график можно в папке encoderProject-master\Saved models\Graphs. Выглядит он следующим образом:

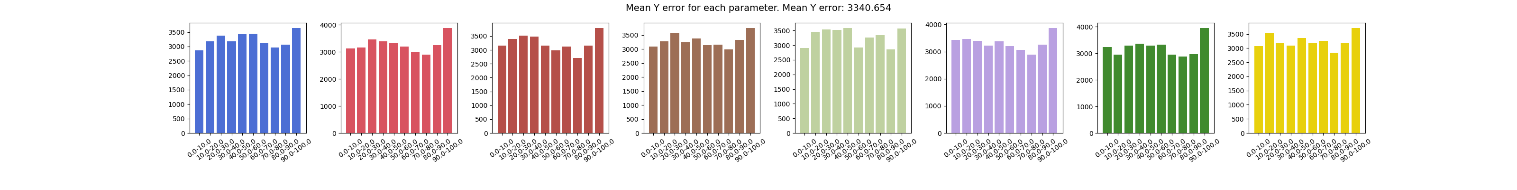


Рисунок 5.

## Проверка корректности исходных данных

В случае возникновения ошибки система выводит на консоль сообщение об этом. В сообщении указывается информация о характере нарушения и местоположение ошибки. Пример обнаружения ошибки приведен на рис. 6.

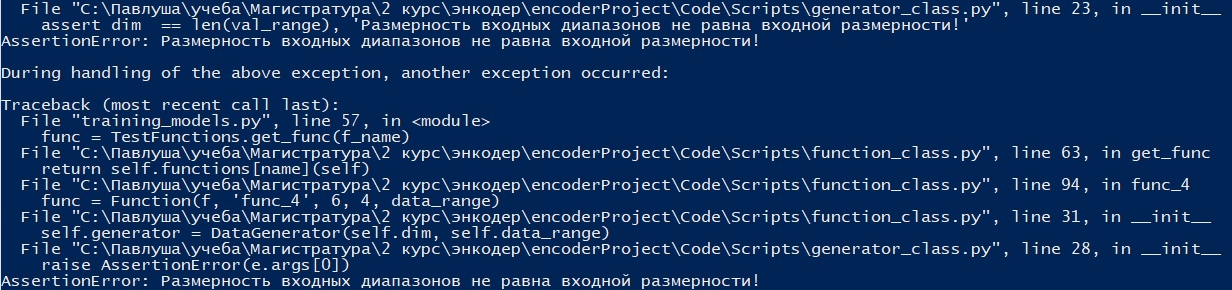


Рисунок 6. Запись об ошибке

# **СООБЩЕНИЯ ОПЕРАТОРУ**

Протокол работы системы содержит:

* сообщения об ошибке в работе программы (рис. 6)
* сообщения о результатах работы программы по тренировке нейронной сети (рис. 1)
* сообщения о результатах работы программы по запуску нейронной сети (рис. 4)