МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Институт №8 «Информационные технологии и прикладная математика»

Кафедра 806 «Вычислительная математика и программирование»

**Лабораторная работа №8**

**по курсу «Нейроинформатика»**

**Динамические сети**

Выполнил: Гамов Павел Антонович

Группа: 8О-407Б-18

Преподаватель: Аносова Н. П.

Москва, 2022

**Условие**

Целью работы является исследование свойств некоторых динамических нейронных сетей,

алгоритмов обучения, а также применение сетей в задачах аппроксимации функций и распознавания динамических образов.

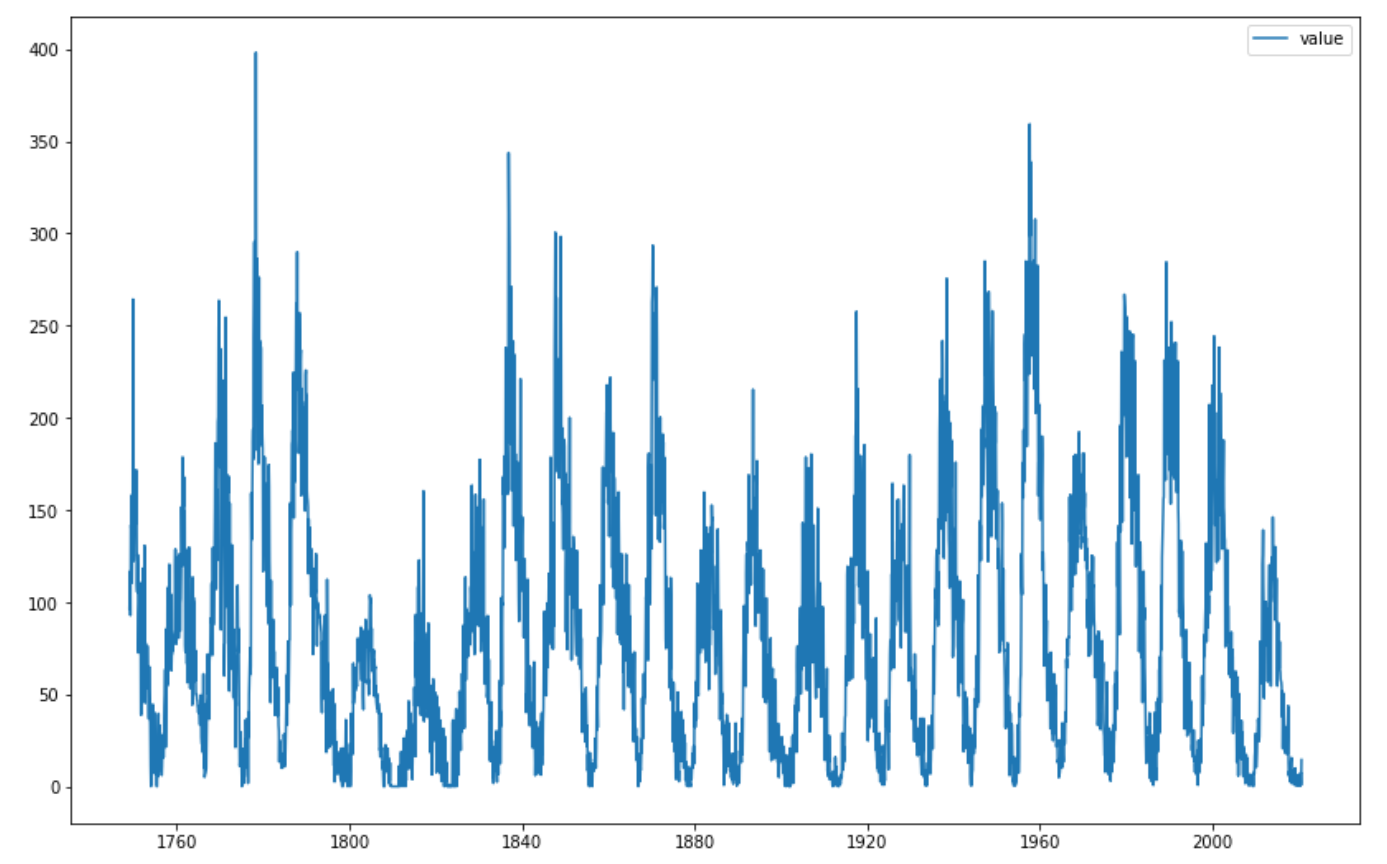
**Решение**

Изображение выглядит как текст

Автоматически созданное описание

**Задание 1**

Построить и обучить сеть прямого распространения с запаздыванием (Focused Time-DelayNeural Network, FTDNN), которая будет аппроксимировать последовательность чисел Вольфа, а также выполнить многошаговый прогноз.



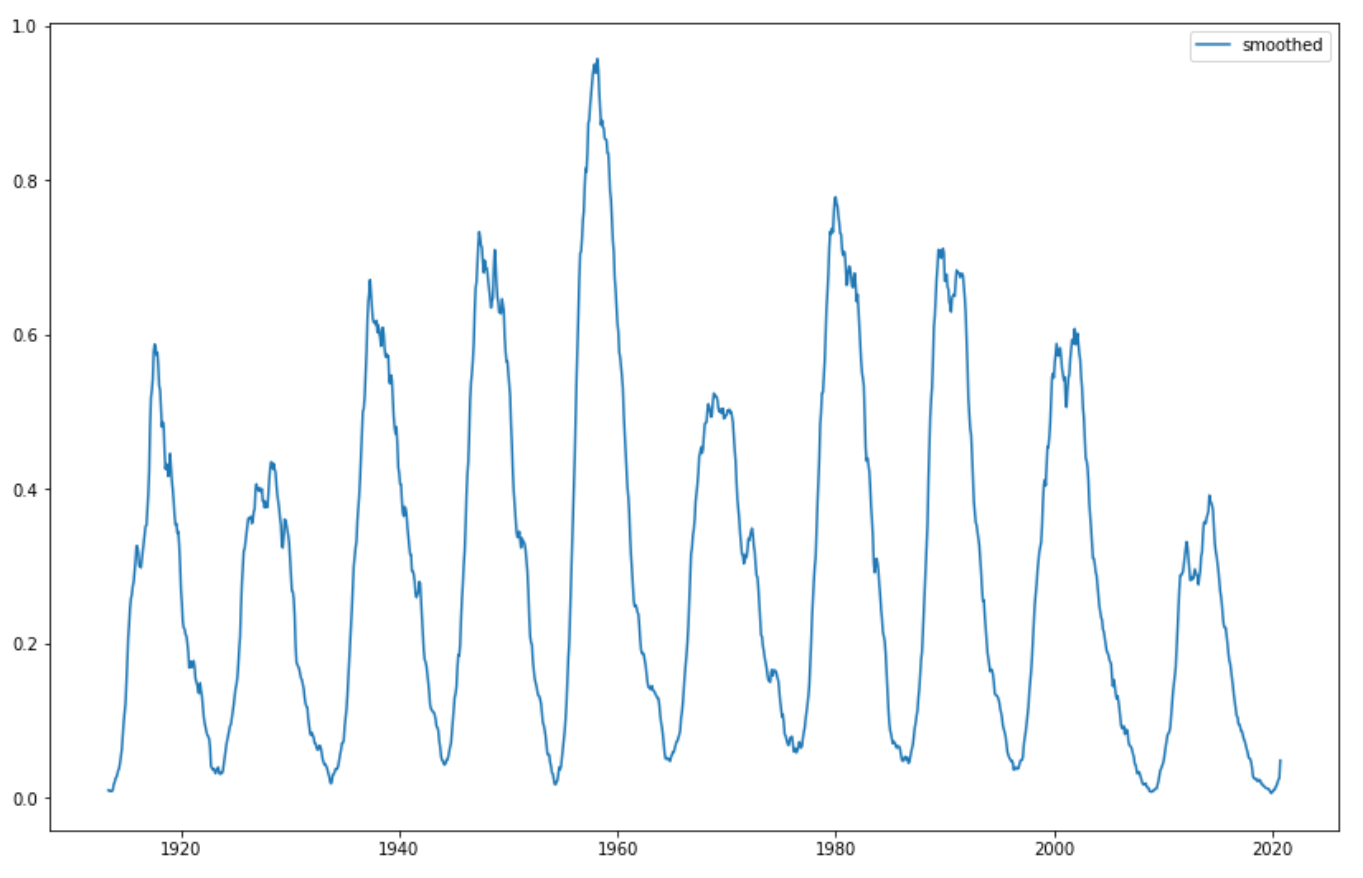
values = df.values.flatten()

Усредняющий фильтр smooth с заданной шириной окна:

Изображение выглядит как текст

Автоматически созданное описание

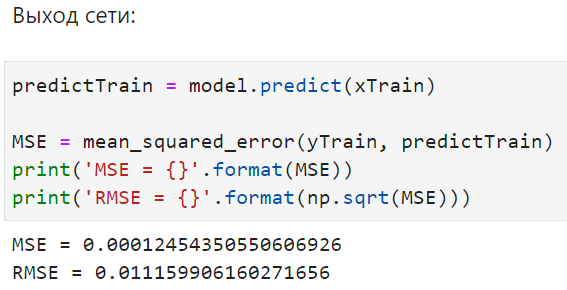
Обновим данные в датафрейме на сглаженные и зададим начало в соответствии с вариантом.

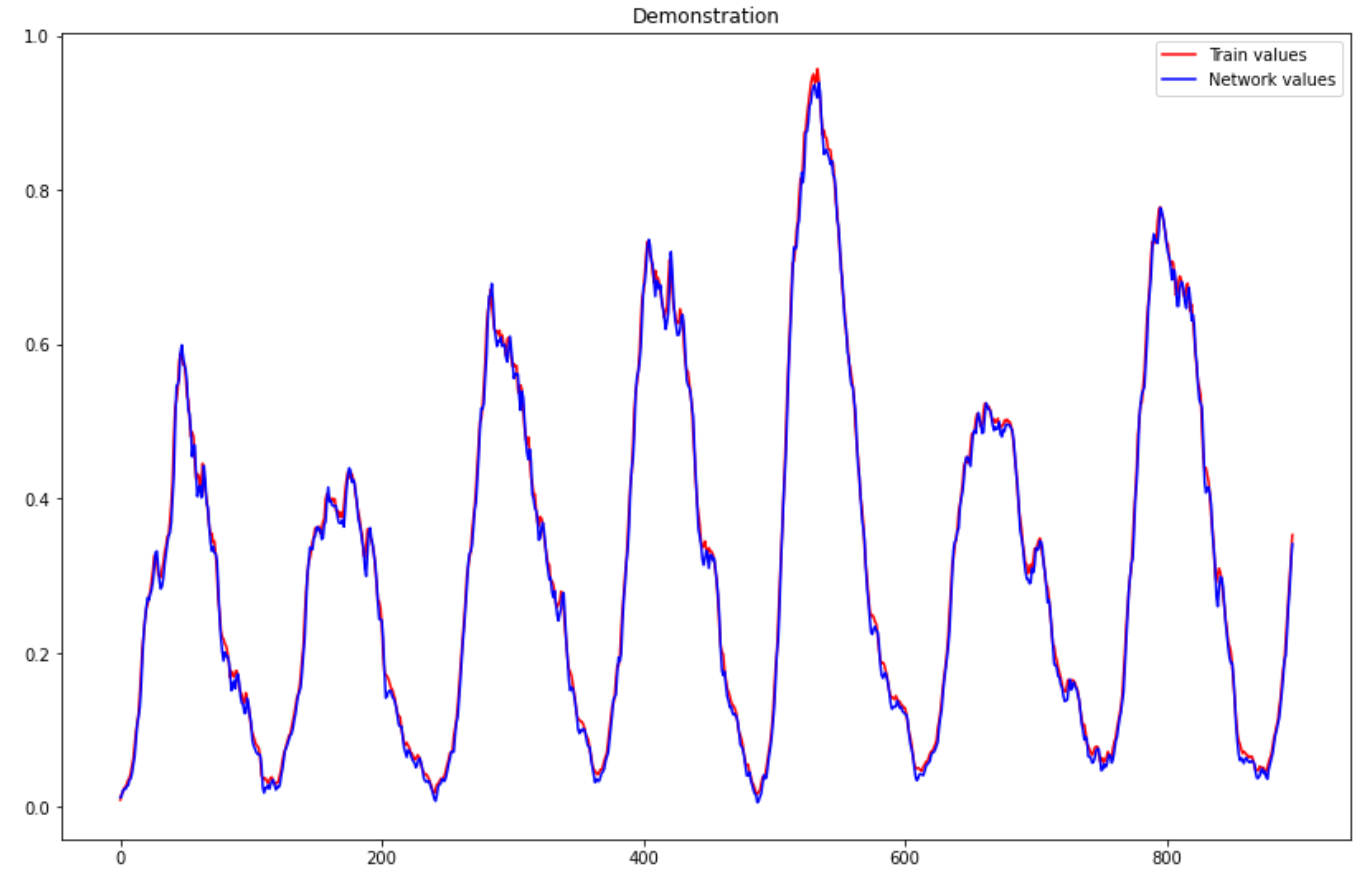


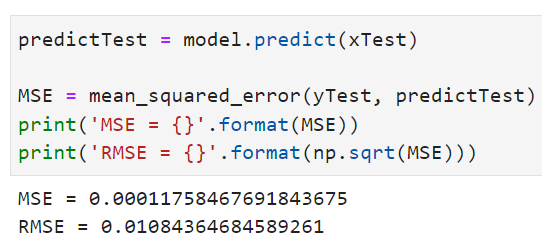
Сформируем обучающее множество (сразу с задержками) и тестовое множество для многошагового прогноза.

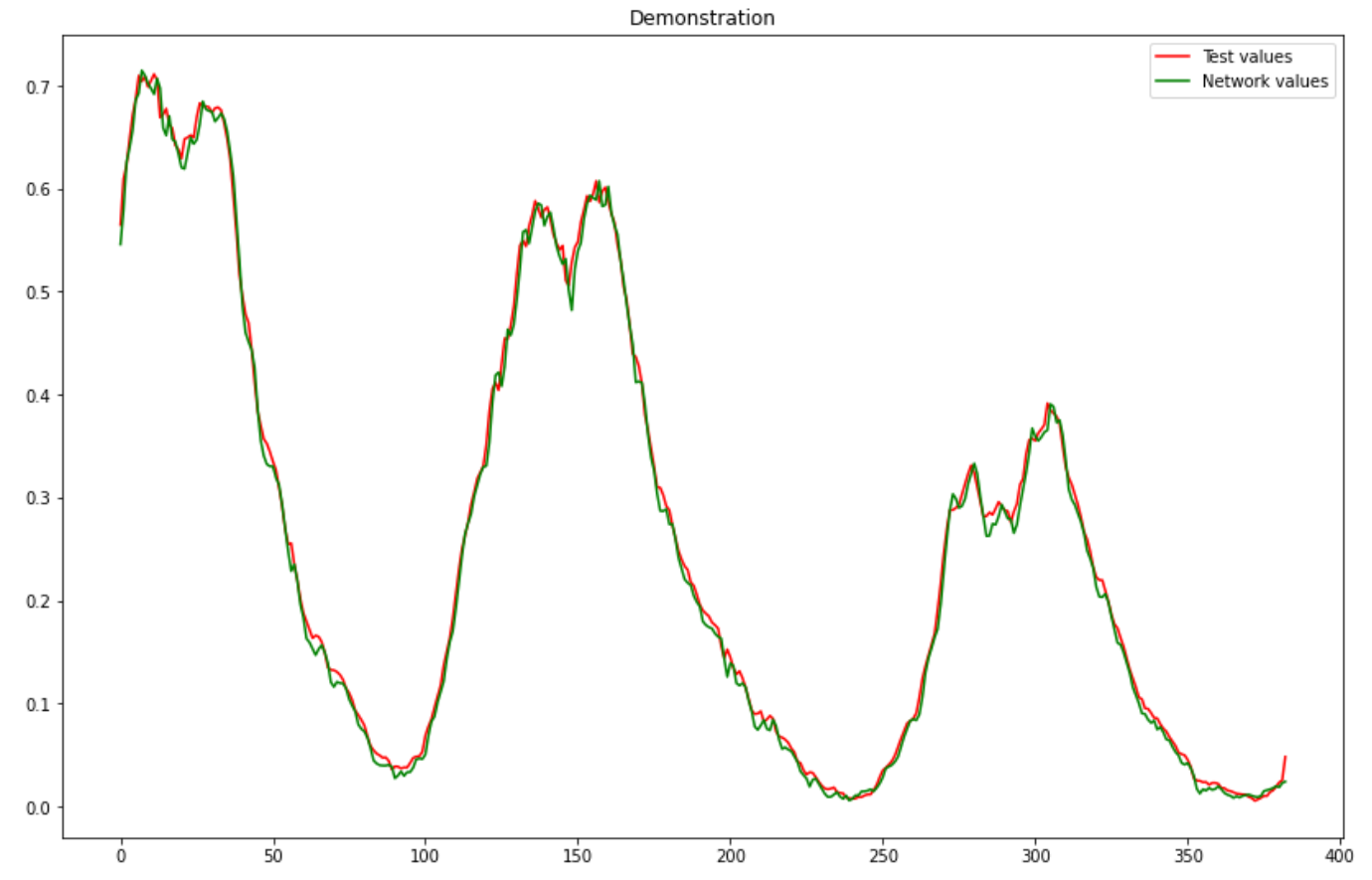
Изображение выглядит как текст

Автоматически созданное описание









**Задание 2**

Построить и обучить сеть прямого распространения с распределенным запаздыванием (Distributed Time-Delay Neural Network, TDNN), которая будет выполнять распознавание динамического образа.

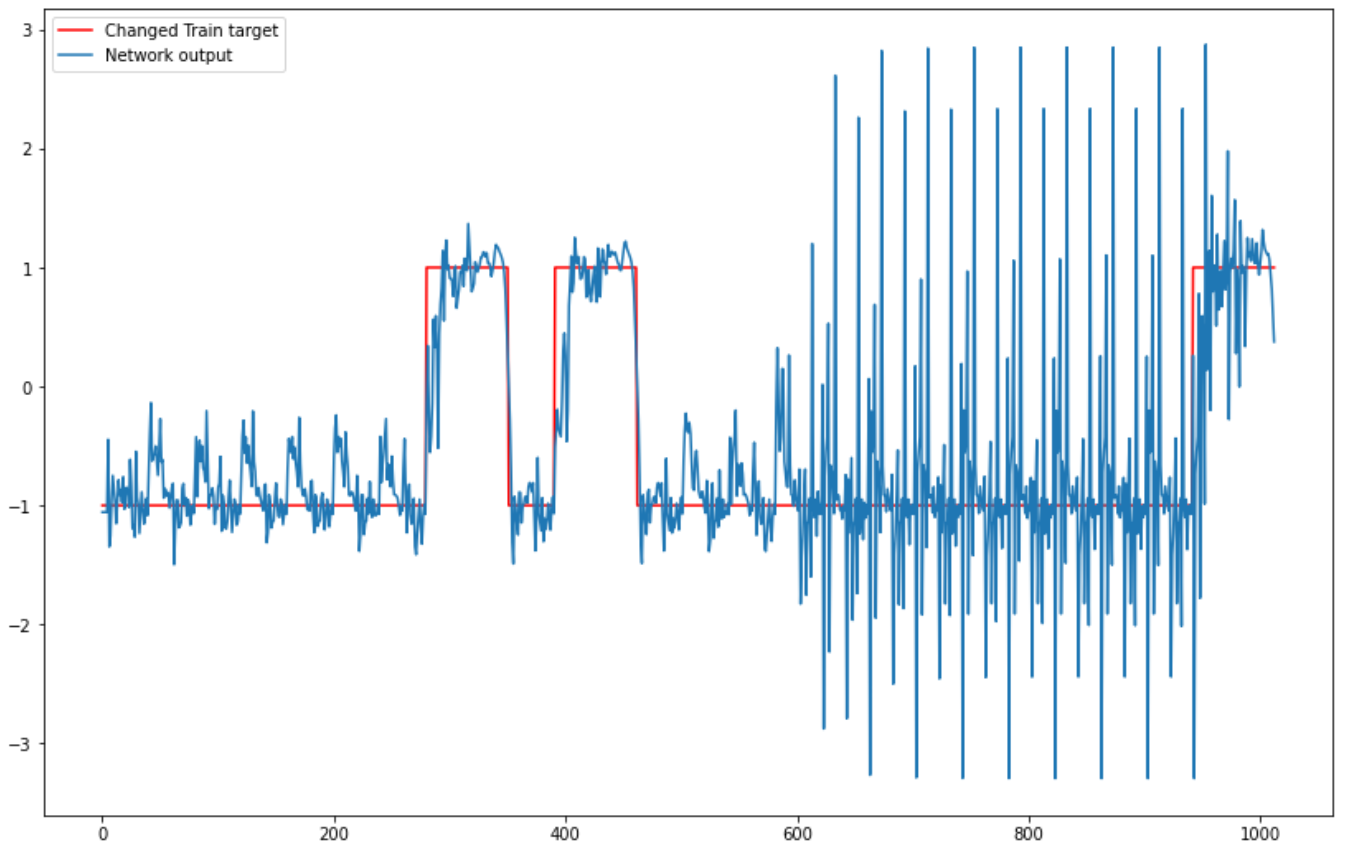
Изображение выглядит как текст, игла

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Для проверки качества распознавания сформируем новое обучающее множество, изменив одно из значений 𝑅.



**Задание 3**

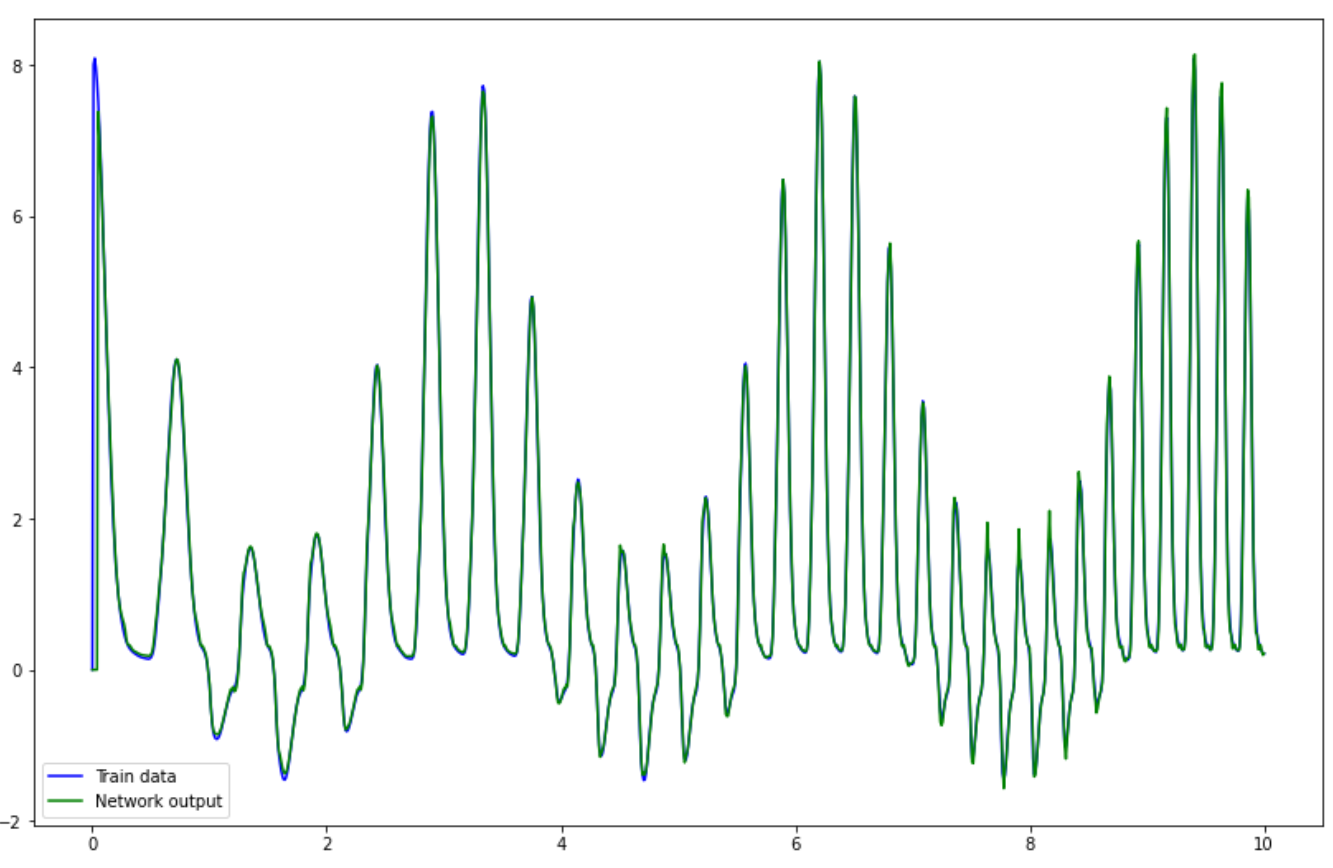
Построить и обучить нелинейную авторегрессионную сеть с внешними входами (Non-linearAutoRegressive network with eXogeneous inputs, NARX), которая будет выполнять аппроксимациютраектории динамической системы, также выполнить многошаговый прогноз значений системы.

narx = NARX(MLPRegressor(hidden\_layer\_sizes=(10, 10)), solver='lbfgs', max\_iter=600,auto\_order=2, exog\_order=[2], exog\_delay=[delay])

narx.fit(inpt, target)

Изображение выглядит как текст

Автоматически созданное описание



Выполним многошаговый прогноз: обучим и рассчитаем выход сети для тестового подмножества.

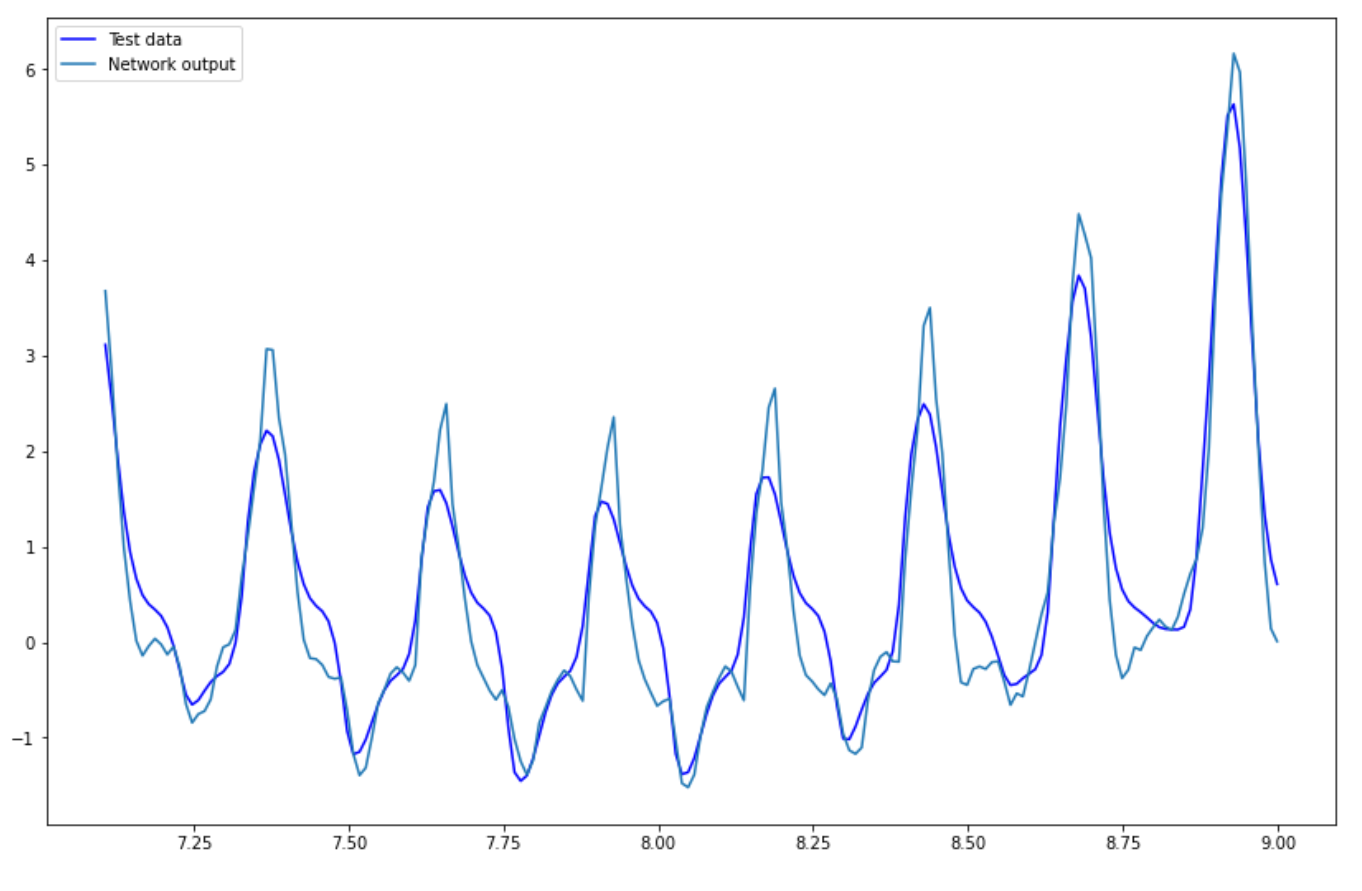
narx = NARX(MLPRegressor(hidden\_layer\_sizes=(10, 10)), solver='lbfgs', max\_iter=600,

auto\_order=2, exog\_order=[delay], exog\_delay=[delay])

narx.fit(inpt, target)

Изображение выглядит как текст

Автоматически созданное описание



**Выводы**

Наиболее интересными из динамических сетей для меня кажутся рекуррентные. С их помощью были созданы такие сети как GPT или BERT которые способны делать то, что ранее, казалось, способны сделать только люди. Если говорить о результатах заданий, то в первом мы снова видим "запоминание" промежутков, как в 4-й ЛР.