ИНТЕЛЛЕКТУАЛЬНЫЕ МЕТОДЫ В ЗАДАЧАХ ЗАЩИТЫ ОТ КИБЕРУГРОЗ

ЛАБОРАТОРНАЯ РАБОТА № 3
Введение в основы работы с библиотекой Keras.

Выполнил:
Мосолков Е.Н.
Преподаватель:
Петров А.А.

Москва 2021 г.

ЦЕЛЬ РАБОТЫ

Знакомство с возможностями библиотеки Keras на примере простого прогнозирования

ОПИСАНИЕ ЗАДАЧИ

Создать НС с математической функцией, которая будет аналогично примерам из лекций предсказывать следующие значения последовательности (или просто конкретные значения). Креативность, сложность функций, а также степень глубины анализа параметров НС – поощряются.

КОД И ОПИСАНИЕ ЕГО КЛЮЧЕВЫХ ВЕЩЕЙ

```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow import keras
from tensorflow.keras.layers import Dense

a = np.array([1, 2, 3, 4, 5, 6, 7,8, 9, 10, 11 ])
b = np.array([6, 10, 14, 18, 22, 26, 30,34, 38, 42, 46])
model = keras.Sequential()
model.add(Dense(units=1, input_shape=(1,), activation='linear'))
model.compile(loss='mean_squared_error', optimizer=keras.optimizers.Adam(0.1))
history = model.fit(a, b, epochs=500, verbose=0)
print("Обучение завершено")
print(model.predict([12]))
print(model.get_weights())
plt.plot(history.history['loss'])
plt.grid(True)
plt.show()
```

ОТЧЕТ

Подавая на вход 2 ряда и пройдя 500 эпох получим результаты (Результат на рис 1.):

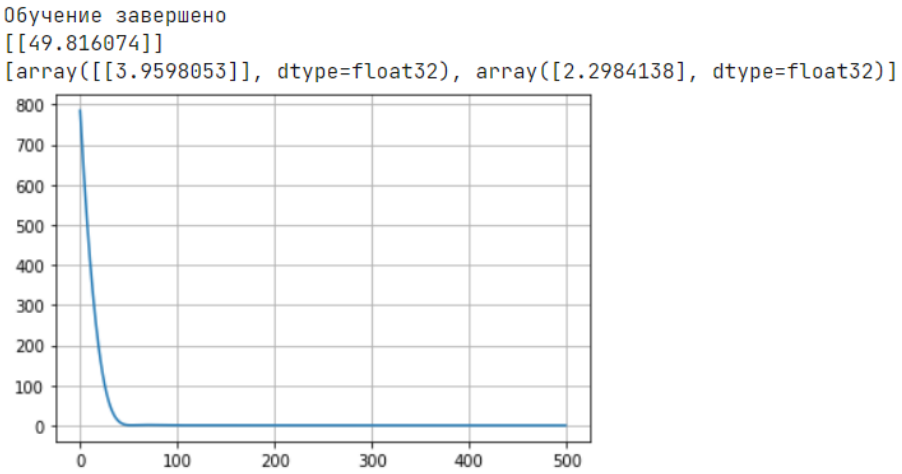

Рисунок 1

Нейросеть определила, что при значении первого ряда = 12, соответствующее значение второго ряда будет равно 49.7, что очень близко к сути (12*4+2 = 50). А параметры нашего уравнения 3.93 и 2.47 соответственно (4 и 2). Следует отметить, что экспериментируя с числом эпох и параметрами оптимизации мы можем улучшить наши результаты. Например, изменив параметр на keras.optimizers.Adam на 0.5 (шаг градиентного спуска) мы получим значения (Рис.2):

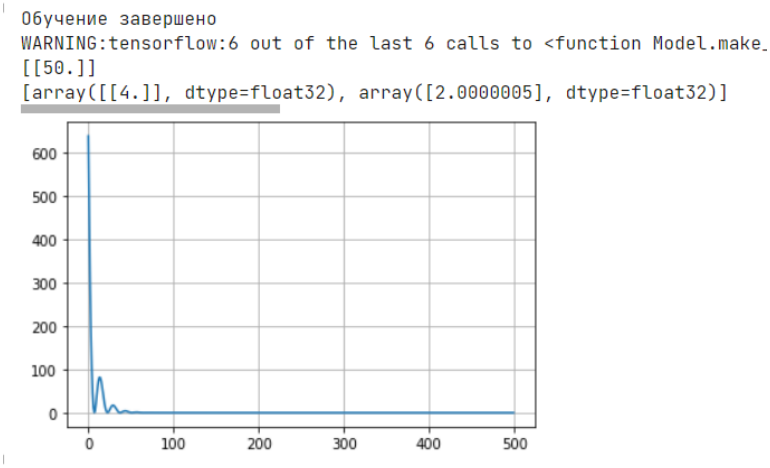

Рисунок 2

Что максимально близко к правильным, однако уменьшая шаг до 0.01 эффект будет обратный (Рис. 3):

```
Обучение завершено
[[40.933365]]
[array([[3.1375868]], dtype=float32), array([3.2823236], dtype=float32)]
```

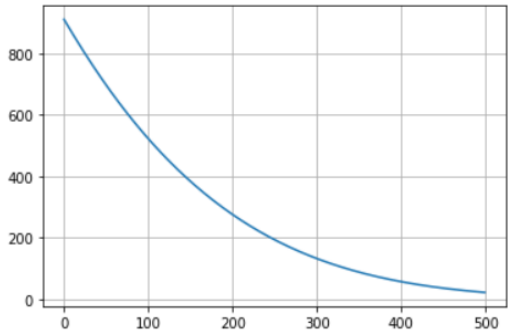

Рисунок 3

Точность существенно снизится. Таким же образом можно «играться» с количеством эпох. Оставим значение шага спуска по умолчанию на 0.1 как и было изначально, но изменим число эпох с 500 на 100 (Рис. 4).

```
Обучение завершено
[[49.16344]]
[array([[3.7323918]], dtype=float32), array([4.374736], dtype=float32)]
```

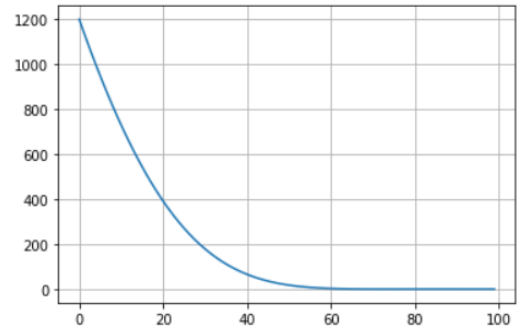

Рисунок 4

Что абсолютно не соответствует действительности и очень далеко от истины. Теперь, наоборот, увеличим число эпох до 10000 (рис. 5):

```
Обучение завершено
[[49.98577]]
[array([[3.9968908]], dtype=float32), array([2.0230825], dtype=float32)]
```

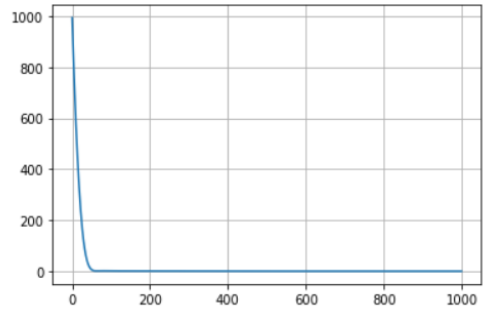

Рисунок 5

Результат снова очень близок к корректному. Таким образом, настоящая лабораторная демонстрирует факт того, что параметры НС часто подбираются опытным путём и для корректной работы НС необходима серьёзная апробация.

ВЫВОД

Я познакомился с возможностями библиотеки keros на python на примере простого прогнозирования