**1. Постановка задачи**

Источник: https://opendata.mkrf.ru/opendata/7705851331-stat\_museum

Данные о музеях России: размер коллекции, график работы и т.п.

Структура:

name - Название музея

base\_fond - Число предметов основного фонда на конец года

science\_fond - Число предметов научно-вспомогательного фонда на конец года

ams - Автоматизированная музейная система: число внесенных музейных предметов

n\_buildings - Число строений

n\_days - Число дней в году, открытых для посещения

guests - Число посещений выставок, экспозиций + экскурсионных посещений, тысяч человек

employees - Численность работников, человек

Задача: вычисление значения параметра guests на основании остальных параметров.

**2. Подход к обработке данных.**

Name – было использовано прямое унитарное кодирование для перевода текса в цифры. К данным примел функцию MinMaxScaler().

base\_fond – Данные были неравномерно распределены, к данным был применен логарифм. Получилось почти нормальное распределение, поэтому к данным применил функцию StandardScaler().

science\_fond – в данных имели выбросы, решил это проблему “отрезав” выброс. Распределение неравномерное и в данных имелось большое количество нулей. Из-за нулей нельзя применить логарифм, был применен квадратный корень. После все манипуляций применил формулу MinMaxScaler().

Ams – в данных был выброс в районе 300000, выброс был срезан. Распределение неравномерное, большое количество нулей. Применил квадратный корень и функцию MinMaxScaler().

n\_buildings – В данных были выбросы, поэтому данные были “срезаны”. Распределение неравномерное, большое количество нулей. Применил квадратный корень и функцию MinMaxScaler().

n\_days – Данные были распределены не равномерно, использовал логарифм и MinMaxScaler().

Guests – В данных были выбросы, выбросы “обрезал”. Применил логарифм. Данные были почти близки к нормальному распределению, но применил MinMaxScaler().

**3.Базовые параметры нейронной сети.**

Количество нейронов входного слоя – 7

Первый скрытый слой – 70

Второй скрытый слой – 15

Третий скрытый слой – 40

Выходной слой – 1

Количество эпох – 100

Батчей - 16

**4. Описание проведенных экспериментов.**

Первая модель работал корректно, хотел увеличить точность.

Второй модели добавил 4 скрытый слой с количеством нейронов 80 и уменьшил количество нейронов в 3 слое до 15. Модель не успевала достаточно обучиться за 100 эпох. Увеличил количество эпох до 150. Модель стала выдавать более точный результат примерно на 0,4%. Количество параметров для обучения стало больше. Нейронная сеть стала сложнее. На тренировочной выборке модель работает лучше, чем тестовой.

В третьей модели решил оставить 2 скрытых слоя. В первом скрытом слое 70 нейронов, во втором скрытом слое 50 нейронов. Количество эпох 110.

Сеть стала ощутима медленнее, использует в 2 раза больше параметров, чем 1 модель. Точно почти не отличается. На тренировочной выборке модель работает лучше, чем тестовой.

**5. Итоговый вывод.**

В моем случае лучше всего использовать три скрытых слоя. Два слоя слишком мало, модель обучается не “плавно” Требует большего количество эпох и параметров. Если слоев слишком много то их влияние на сеть слабеет, есть несколько слоев, которые больше всего влияют на результат нейронной сети.

Лучше чередовать количество нейронов в скрытых слоях. Если мы уменьшим количество нейронов в следующем слое, тогда нейронная сеть попробует обобщить информацию и найти новые закономерности. Если нам нужно более тщательно обработать информацию тогда мы должны увеличь число нейронов в следующем слое.