

УДК 004.89

ПРОБЛЕМАТИКА ФОРМИРОВАНИЯ ОБОБЩАЮЩЕЙ ФУНКЦИИ ДАНЫХ В АЛГОРИТМАХ МАШИННОГО ОБУЧЕНИЯ РЕГРЕССИ- ОННОГО ТИПА

Селезнева Е. Ф., ст. группы ИТм-211

Научный руководитель: Протоdjяконов А. В., к.т.н., доцент

Кузбасский государственный технический университет
имени Т.Ф. Горбачева

В среде прикладного искусственного интеллекта широкое распространение получили искусственные нейронные сети. Так сложилось не случайно: эти модели [нейронные сети], являющиеся представителями класса глубокого обучения, позволяют решать задачу предметной области таким образом, что при появлении новых прецедентов в наборе анализируемых признаков, алгоритм решения остается устойчивым [1]. То есть, он способен к самостоятельному «автоматическому» обучению в процессе решения задачи, не требуя вмешательства разработчиков. Это коренное отличие отделяет модели глубокого обучения от всего класса алгоритмов машинного обучения [2].

Одной из важнейших описательных характеристик нейронных сетей является их полнота [2, 3].

Под полнотой подразумевается формирование такой обобщающей данные функции, которая могла бы не просто экстраполировать значения, а полноценно описывать закон предметной области. Именно это качество и позволяет в дальнейшем составлять более точные прогнозы модели на неизвестных ранее данных прецедентов.

Формализуем теоретические положения на языке математических формул. Для простоты рассмотрим наиболее простой тип нейронной сети – двухслойную искусственную нейронную сеть, состоящую из входного и выходного слоя соответственно.

В качестве основы рассмотрим выборку данных (1), репрезентативно отражающую генеральную совокупность некоторой предметной области.

$$X^l = (x_i; y_i)_{i=1}^l \quad (1)$$

В обучающей выборке (1) x_i являются объектами обучающей выборки, причем $x_i \in R^n$, а y_i – ответами выборки.

Выделяют два типа задач:

- Задача классификации (когда предсказывается принадлежность объекта обучающей выборки к заданному целевому классу);
- Задача регрессии (когда прогнозируется число: например, *вероятность* отнесения объекта к заданному классу).

Для задачи регрессии $Y = R$, то есть ответами являются действительные числа. Линейная модель регрессии обобщенно может быть записана в виде (2).

$$a(x, \omega) = \langle \omega, x_i \rangle \quad (2)$$

Для задачи дихотомической классификации линейная модель классификатора будет оформлена в виде формулы (3).

$$a(x, \omega) = \text{sign}\langle \omega, x_i \rangle \quad (3)$$

В качестве невозрастающей функции отступа может быть использована, например, функция $\ln(1 + e^{-M})$, $(1 - M)_+$, e^{-M} и другие.

И та, и другая задача (регрессии и классификации) сводятся к однотипным задачам оптимизации [2].

Однотипность означает, что функционал представляет собой сумму по всем объектам выборки от некоторой функции потерь, которая зависит от вектора весов. Такой тип функционала позволяет использовать методы стохастического градиента и другие элементы оптимизации нейронных сетей.

Данная концепция является одним из основных принципов построения нейронных сетей, которые различаются между собой лишь способом организации внутренних (скрытых) слоев, функцией отступа и способом оптимизации функционала.

Вторая базовая концепция работы нейронных сетей основана на понятии нейрона. Нейрон на входе принимает заряды. Заряды внутри клетки накапливаются, и клетка выдает импульс, когда накопленный заряд превышает некий порог – эти простые нейрофизиологические знания привели исследователей к тому, что математическая линейная модель может очень приближенно описывать функционирование нервной клетки (рисунок 1).

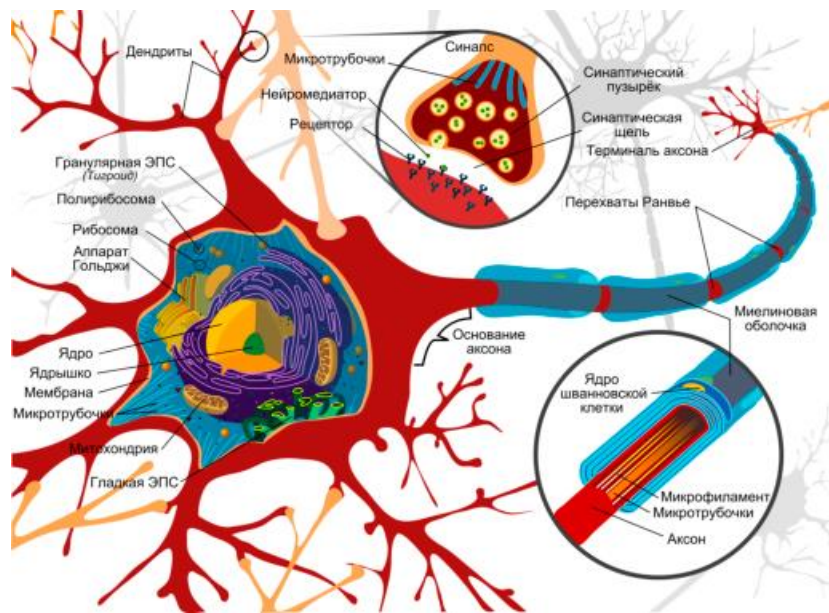


Рисунок 1 – Структура и функционирование нервной клетки

Поскольку заряд есть величина аддитивная, то в модели двуслойной (и сетей с большим количеством слоев) нейронной сети они [заряды] заменены на веса, настройка которых и позволяет решать поставленную перед алгоритмом задачу [3].

Математическим языком эта модель описывается формулой модели двуслойной нейронной сети (4).

$$a(x, \omega) = \sigma(\langle \omega, x \rangle) = \sigma \left(\sum_{j=1}^n \omega_j f_j(x) - \omega_0 \right) \quad (4)$$

В формуле (4) $f_j: X \rightarrow R, j = 1, \dots, n$ – числовые признаки; ω_j – веса признаков, $\sigma(z)$ – функция активации.

Модель (4) может представлена в схематичном функционировании как обобщающая (и суммирующая) функция значений весов (рисунок 2).

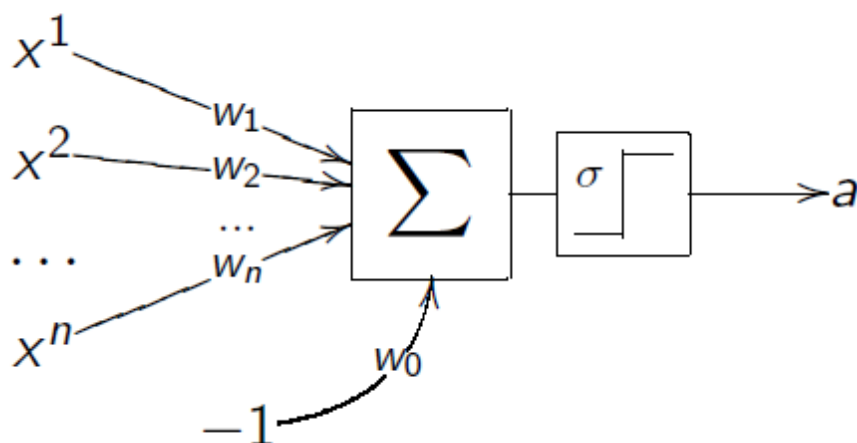


Рисунок 2 – Схематичное представление функционала однослойного персептрона

Отметим, однако, что естественные нейронные сети, эволюционируя, пришли к тому, что стали сетями [4]. Это обосновано тем, что отдельная нервная клетка может выполнять (исчислять) только очень примитивную функцию.

В задачах прикладного искусственного интеллекта, фактически, один нейрон реализует одну разделяющую линейную гиперплоскость в пространстве векторов X . Очевидно, что единичный нейрон не в состоянии аппроксимировать любую функцию, так как линейные функции – это довольно узкий класс в пространстве всех функций.

Между тем, достичь высокой обобщающей способности (полноты) представляется возможным только тогда, когда математическая аппроксимирующая функция алгоритма наиболее точным образом описывает распределенные в N – мерном пространстве признаки.

Таким образом, двухслойная нейронная сеть, которая состоит из слоев входа и выхода не способна полноценно восстановить функцию любого вида (формализованный математический закон данных некоторой рассматриваемой задачи), то есть отсутствует возможность аппроксимировать непрерывную произвольную функцию [5]. Исходя из этого, можно сделать вывод о том, что изучение двухслойной нейронной сети позволяет изучить логику функционирования нейронных сетей в целом, а также понять функционал их работы на атомарном уровне. Однако, решение сложной прикладной задачи, в которой количество i прецедентов x_i более трех, не позволит сформировать надежную функцию обобщающей способности модели глубокого обучения, так как этому противоречат математические законы, на которых основан сам алгоритм [1, 6, 7].

Список используемых источников:

1. Томас Кормен, Чарльз Лейзерсон. Алгоритмы: построение и анализ, 3-е издание – М.: ООО И.Д. Вильямс. 2013. – 1328 с.
2. Yinyan Zhang. Deep Reinforcement Learning with Guaranteed Performance. – Springer Press. 2020. – 265 с.
3. Майкл Солтис. Введение в анализ алгоритмов – СПб.: ДМК. 2019. – 269 с.
4. Дж. Клейнберг, Е. Тардос. Алгоритмы: Разработка и применение – СПб.: Питер. 2016 – 800 с.
5. Allen Downey, Jeffrey Elkner, Chris Meyers. How to Think Like a Computer Scientist - Green Tea Press, 2008. – 250 с.
6. Andreas C. Miller, Sarah Guido. Introduction to Machine Learning with Python - O'Reilly Media. 2017. – 367 с.
7. Орельен Жерон. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow - O'Reilly Media. 2018. – 662 с.