

ЭФФЕКТИВНОЕ ВСТРОЕННОЕ АНСАМБЛИРОВАНИЕ НЕЙРОННЫХ СЕТЕЙ

Тишковский М.А., Лимонтов А.С., Евжик Д.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Чернявский Ю.А. – к.т.н., доцент

Глубокие нейронные сети с большим количеством параметров являются очень мощным семейством алгоритмов машинного обучения. Тем не менее переобучение является серьезной проблемой для таких сетей. Одним из методов борьбы с проблемой переобучения является техника ансамблирования, однако она требует больше количество ресурсов, в том числе и временных. В данном исследовании изучался способ использования техники ансамблирования внутри нейронной сети путем исключения части нейронов из нейросети во время стадии обучения.

Глубокие нейронные [1] сети содержат огромное количество скрытых слоев с нелинейными функциями активации, что позволяет им выучить очень сложную функцию отображения входных данных на целевое значение. Поскольку количество данных зачастую сильно ограничено, настраиваемые коэффициенты нейронной сети могут подстроиться под тренировочные данные, однако выдавать плохие результаты на реальных данных, распределение которых будет хоть немного, но отличаться от распределения тренировочных данных. Для борьбы с эффектом переобучения на тренировочных данных изобрели множество методов, например, регуляризация [2], аугментация данных [3], ансамблирование [4] нескольких моделей.

Идея ансамблирования заключается в том, что независимо тренируется несколько нейронных сетей, после этого для финального предсказания используется усреднение предсказаний всех моделей. Математическое обоснование данного метода заключается в том, что математическое ожидание ошибки ансамбля моделей будет меньше ошибки отдельно взятой модели нейронной сети. Лучших результатов можно добиться, если архитектуры нейросетей будут отличаться друг от друга или будут натренированы на разных данных. Нахождение оптимальных гиперпараметров для каждой сети является достаточно сложной задачей, а сама тренировка множества моделей очень ресурсоемкая операция. Более того, данных может быть недостаточно для того, чтобы натренировать несколько нейронных сетей на разных подмножествах исходных тренировочных данных.

Исключение части нейронов из нейронной сети во время стадии обучения решает обе описанные выше проблемы. Данная техника предотвращает переобучение и предоставляет способ приблизительного комбинирования экспоненциального количества различных нейронных сетей вычислительно эффективным способом. Опишем более подробно работу алгоритма на стадиях обучения и тестирования.

На стадии обучения при прямом проходе слой исключения части нейронов зануляет коэффициенты нейронов по схеме Бернулли [5] с вероятностью p , где p – гиперпараметр данного слоя. После этого полученные коэффициенты делятся на $(1-p)$, чтобы сумма коэффициентов не менялась. Соответственно при обратном проходе с помощью алгоритма обратного распространения ошибки исключенные нейроны не участвуют в обновлении коэффициентов.

Поскольку при каждом прямом проходе нейросети на стадии обучения случайным образом исключаются некоторые нейроны, нейронная сеть может иметь 2^n состояний, что ведет к аппроксимации ансамбля моделей. Также поскольку для тренировки используется мини-батч [6] техника, где данные для итерации подаются подмножеством исходных данных и коэффициенты нейронной сети обновляются на основе градиента данной мини-батча, то ансамблируются модели, натренированные на различных данных.

Прямой проход стадии тестирования не исключает никаких нейронов и не делит ни на какие коэффициенты.

Таким образом, мы описали алгоритм, который вычислительно эффективно аппроксимирует технику ансамблирования нейросетей, обученных на разных данных.

Список использованных источников:

1. Deep learning [Электронный ресурс] – Режим доступа: https://en.wikipedia.org/wiki/Deep_learning
2. L1 и L2 регуляризация для линейной регрессии [Электронный ресурс] – Режим доступа: <https://craftappmobile.com/l1-vs-l2-regularization/>
3. [Электронный ресурс] – Режим доступа: <https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced>
4. Two is better than one: Ensembling Models [Электронный ресурс] – Режим доступа: <https://towardsdatascience.com/two-is-better-than-one-ensembling-models-611ee4fa9bd8>
5. Bernoulli scheme [Электронный ресурс] – Режим доступа: https://en.wikipedia.org/wiki/Bernoulli_scheme
6. A Gentle Introduction to Mini-Batch Gradient Descent and How to Configure Batch Size [Электронный ресурс] – Режим доступа: <https://machinelearningmastery.com/gentle-introduction-mini-batch-gradient-descent-configure-batch-size/>

ОЦЕНКА АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ. ДИЛЕММА СМЕЩЕНИЯ-ДИСПЕРСИИ

Тишковский М.А., Лимонтов А.С., Подвальников Д.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Чернявский Ю.А. – к.т.н., доцент

В данной работе рассмотрена оценка алгоритмов машинного обучения, а также дилемма смещения-дисперсии. Были проведены эксперименты по оценке различных алгоритмов машинного обучения через аппроксимацию функции косинуса, результаты которых представлены в работе.

В настоящее время с помощью алгоритмов машинного обучения решают все большее количество задач. В связи с этим возникает проблема, как правильно подобрать алгоритм машинного обучения под конкретную задачу, а также как оценить полученную натренированную с помощью этого алгоритма модель. Вторая задача решается с помощью введения правильных метрик оценки качества, что само по себе тоже является непростой задачей.

Задача выбора алгоритма машинного обучения является достаточно сложной и требует анализа исходных данных, понимания доменной области, знаний особенностей семейств алгоритмов. В связи с этим вводится понятие дилеммы смещения-дисперсии [1], по которому модели с меньшим смещением в параметре оценки имеют более высокую дисперсию, и наоборот, что в терминах машинного обучения является проблемой недообученности или переобученности алгоритма.

Переобучение [2] – явление, когда построенная модель хорошо объясняет примеры из обучающей выборки, но относительно плохо работает на примерах, не участвовавших в обучении.

Недообучение случается, когда алгоритм машинного обучения не может найти закономерности между исходными данными и целевой зависимостью.

Рассмотрим задачу аппроксимации функции косинуса на промежутке от 0 до 2π с помощью различных алгоритмов. На рисунке 1 представлена аппроксимация с помощью линейной регрессии [3], на рисунке 2 с помощью метода опорных векторов [4] с ядром радиальной базисной функции с параметром $\gamma=1$, на рисунке 3 с помощью метода опорных векторов с ядром радиальной базисной функции с параметром $\gamma=100$.

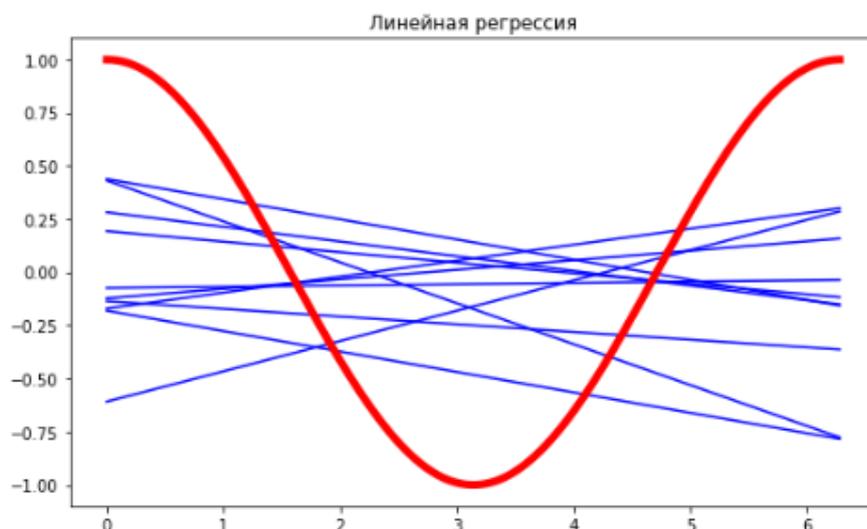


Рисунок 1 – Аппроксимация функции $\cos(x)$ с помощью линейной регрессии