

1

Здравствуйте, меня зовут Гузенко Алексей, я учусь на факультете компьютерных наук, по специальности математика и компьютерные науки и сегодня в вам расскажу о вероятностном подходе к обработке последовательных данных.

2

Компании на данные момент собирают огромное количество данных, помимо того, что из них нужно вычлениить нужную нам информацию, структурировать ее и сделать к ней нужны предположения, ее нужно будет в дальнейшем скормить модели машинного обучения. Так как рост интернет пользователей растет с каждым годом, а вычислительные мощности не поспевают за этим ростом, стоит вопрос поиска новых, более оптимальных алгоритмов машинного обучения, как раз мы рассмотрим один из них, это вероятностный подход в машинном обучении.

3

Для этого мы рассмотрим применение теоремы Байеса в машинном обучении, в частности в нейронных сетях, посмотрим на современные алгоритмы поиска нужного вероятностного распределения для модели, оценим результат работы алгоритмов и посмотрим на сравнение обычного подхода с вероятностным

4

Нужно начать с самой теоремы Байеса.

Априорная вероятность должна соответствовать тому, что мы знаем о вероятностном распределении перед рассмотрением данных.

Апостериорная вероятность результат байесовского анализа, который отображает все, что известно о задаче с учетом имеющихся данных и используемой модели

Правдоподобие – данные с учетом принятых параметров

Предельное правдоподобие – вероятность исследуемых данных, усредненная по всем возможным значениям, которые могут принимать параметры. Обычно не уделяют особое внимание

5

Тут можно наглядней увидеть работу Байесовской теоремы. Самое правое вероятностное распределение, это вероятность нашей модели. По центру результат Байесовской теоремы, слева правдоподобие.

6

Помимо разных моделей по типу Байесовских классификаторов, которые сейчас активно используются в некоторых сферах, мы можем использовать вероятностные распределения в качестве веса в нейронной сети.

7

Для того, чтобы не приходилось подбирать параметры распределения вручную существуют ряд алгоритмов, которые помогут нам в этом. Один из них — это алгоритм No U Turn, что на русский переводит как нет разворота. Мы используем блуждающую точку, и пока она не начнет разворачиваться, то есть пока скалярное распределение между направлением импульса и направление от начальной точки до текущей не станет нулевым или отрицательным, мы выполняем итерацию алгоритма

Для блуждания мы используем метод Стёрмера — Верле с равновозможным выбором одного из направлений.

Так же нам нужно задать целевое распределение, чтобы исключить неоптимальные параметры.

8

Точка на которой мы остановимся и будет нашей апостериорной вероятностью, за исключением тех точек, которые не попадают под наши требования, то есть красные точки и перечеркнутые точки. Перечеркнутые точки не попадают под требование детального равновесия.

9.

Тут можно видеть блок-схему алгоритма. То есть мы задаем целевое распределение, выбираем направление движения, находим новые параметры и новый импульс, проверяем условие. Если условие выполняется, то возвращаемся к выбору направления, если нет, находим новое распределение.

10.

Так выглядит результат работы алгоритма, справа можно увидеть все результаты наших проб, то есть 2000 проб дали нам распределение параметров слева.

11

Так выглядит уже наше искомое апостериорное распределение. Синие линии — это наша выборка, черная линия — распределение данных, оранжевая пунктирная линия — среднее апостериорное распределение

11

Еще один из основных алгоритмов — это алгоритм Вариационного вывода. Тут мы при помощи распределения  $q$  должны найти такие параметры, чтобы распределение апостериорное  $p$  было максимально близко к  $q$ . Для этого используется метрика для определения расстояния между двумя распределениями, относительная энтропия.

12

На графике мы можем увидеть, что с каждой итераций алгоритм подбирает все более оптимальные параметры

13

Давайте рассмотрим работу нейронной сети с вероятностным подходом на определенных данных. Нам нужно выполнить классификацию объектов.

14

Вот как с этим справилась наша нейронная сеть, с точностью около 90 процентов.

15

На данном графике мы можем посмотреть, как наша модель видит данные.

16

Сравнение двух подходов

17

Вывод