**Гиперпараметры нейронной сети**

Гиперпараметры нейронной сети – параметры, которые не настраиваются в процессе обучения, а задаются до процесса обучения человеком вручную. Возникает вопрос, как задать оптимальные значения для конкретной модели?

По сути рабочий метод – перебор:

1. Формируем список возможных значений параметра B
2. Для каждого значения bi случайным образом делим выборку на тренировочную и валидационную
3. Тренируем сеть на тренировочной, считаем точность на валидационной
4. Находим bi, где максимальная точность

Кросс-валидация (скользящий контроль) – разбиваем выборку на n частей, на каждом шаге тренируем сеть на остальных частях, а текущим валидируем, точность – среднее арифметическое

Количество слоев

Традиционно существуют некоторые разногласия по поводу того, как считать количество слоев.

Разногласия связаны с тем, считается ли входной слой. Есть аргумент, чтобы предположить, что его не следует считать, потому что входные данные не активны; они просто входные переменные. Мы будем использовать это соглашение; это также соглашение, рекомендованное в книге «Нейронное кузнечное дело«.

Следовательно, MLP, который имеет входной слой, один скрытый слой и один выходной слой, является 2-уровневым MLP.

Количество нейронов на слой

Сколько нейронов на слое

Скорость обучения

Скорость обучения нейронной сети – один из наиболее важных параметров, контролирующих процесс обучения. Данный параметр определяет величину изменения весовых коэффициентов межнейронных связей. Для идеального приближения к минимуму ошибки нейронной сети скорость обучения должна стремиться к бесконечно малому значению для обеспечения наилучшей сходимости алгоритма обучения. Однако чем меньше выбранное значение шага обучения, тем дольше происходит обучение сети. Таким образом, при стремлении шага обучения к бесконечно малому значению время, требуемое для обучения нейронной сети, многократно возрастает.

Регуляризация L1 L2

Регуляризация - метод добавления некоторых дополнительных ограничений к условию с целью решить некорректно поставленную задачу или предотвратить переобучение. Чаще всего эта информация имеет вид штрафа за сложность модели.

Как говорилось ранее, регуляризация полезна для борьбы с переобучением. Если вы выбрали сложную модель, и при этом у вас недостаточно данных, то легко можно получить итоговую модель, которая хорошо описывает обучающую выборку, но не обобщается на тестовую.

Переобучение в большинстве случаев проявляется в том, что итоговые модели имеют слишком большие значения параметров. Соответственно, необходимо добавить в целевую функцию штраф за это. Наиболее часто используемые виды регуляризации — L1 и L2, а также их линейная комбинация — эластичная сеть.

Тип активации выходного слоя

Функция активации определяет выходное значение нейрона в зависимости от результата взвешенной суммы входов и порогового значения.

Тип функции потери

Эта функция, по сути, вычислит, насколько хорошо работает наша модель, сравнив то, что модель прогнозирует, с фактическим значением, которое она должна выдает. Если Ypred очень далеко от Yi, значение потерь будет очень высоким. Однако, если оба значения почти одинаковы, значение потерь будет очень низким. Следовательно, нам нужно сохранить функцию потерь, которая может эффективно наказывать модель, пока та обучается на Тренировочных данных (Train Data).

Количество эпох

Произошла одна эпоха (epoch) — весь датасет прошел через нейронную сеть в прямом и обратном направлении только один раз. С увеличением числа эпох, веса нейронной сети изменяются все большее количество раз. Кривая с каждый разом лучше подстраивается под данные, переходя последовательно из плохо обученного состояния (последний график) в оптимальное (центральный график). Если вовремя не остановиться, то может произойти переобучение (первый график) — когда кривая очень точно подстроилась под точки, а обобщающая способность исчезла.