семейство алгоритмов, известн.как нейронные сети, недавно возродилось под названием **"глубокое обучение"**

алг.глуб.обуч.жестко привязаны к конкр.случ.исп-ния.

рассм.нек.относ.простые методы, а именно многослойные персептроны для классиф.и регрессии, кот.м.служить отправн.т.в изуч.более сложн.методов

многосл.персептроны (MLP) = простые нейронные сети прямого распространения = нейронные сети

МОДЕЛЬ НЕЙРОННОЙ СЕТИ

MLP м.рассм.как обобщ.лин.моделей, кот.прежде чем прийти к решению вып-ет неск.этапов обработки д-х.

Визуализируем логистическую регресиию [1].

* узлы слева – вх.признаки
* соединительные линии – коэф.
* узел справа – выход (взвешенная сумма входов)

В MLP процесс вычисления взвешенных сумм повторяется несколько раз. Сначала вычисляются **скрытые элементы** (промеж.этап обработки). Они вновь объединяются с помощью взвешенных сумм для получения конечного результата [2].

*Вход – скрытый слой - выход*

У этой модели больше вычисл.коэф(весов):

* коэф.между к.вх и к.скрытым эл-том (кот-е образ.скрытый слой)
* коэф между к.эл-том скрытого слоя и выходом

С мат.т.зр вычисл.взвеш сумм = вычисл 1 взвеш.суммы таким образом, чтобы эта модель обладала более мощной прогнозной силой.

Рассм.пример нейр.сети с 1 скрытым слоем: вх.слой простой передает входы скрытому слою либо без преобраз, либо вып-нив стандартизацию входов. Затем происходит вычисление взвеш.суммы вх.к.эл-та скрытого слоя, а к ней примен.ф-ция активации:

Обычно исп.нелин.ф-ции: **выпрямленный лин.эл-т** или **гиберблический тангенс**

В итоге – выходы нейронов скрытого слоя. Эти выходы стан.входами вых.слоя. Снова вычисл.взвеш.сумму входов, применяем ф-цию активации и получаем итоговые знач.цел.пер.

**Ф-ции активации** [3]:

* relu – отсекает знач.ниже 0
* tanh – приним.знач.от -1 до 1 (для мин и макс знач входов)

М.доб.доп.скрытые слои [4]:

Построение больших нейронный сетей, сост.из мн-ва слоев вычисл, вдохновило специалистов ввести термин **«глубокое обучение»**

НАСТРОЙКА НЕЙРОННЫХ СЕТЕЙ

Посм.как раб.MLP применив MLPClassifier к набору д-х two\_moons [5]:

Как видно на рисунке нейронная сеть построила нелинейную, но отнс.глабку границу прин.реш.

По умолч.MLP исп.100 скрытых узлов, что довольно много для этого неб.набора д-х.

Мы м.уменьшить число (снизить сложность модели) и снова получить хороший рез-т [6]

При исп-нии лишь 10 скрытых элтов граница принятия решений становится неровной. По умолч.исп-ся ф-ция relu.

При исп-нии 1 скрытого слоя ф-ция будет сост.из 10 прямолин.отрезков

Если необх.получить более гладку границу:

* добавить больше скрытых эл-тов, добавить 2ой скрытый слой [7]
* исп-ть ф-цию активации tanh [8]

М.доп.настроить сложность нейронной сети с пом.l2 штрафа, чтобы сжать весовые коэф.до близких к 0 знач.

В MLPClassifier За это отвеч.пар-р alpha и по умолч.устан. низкое знач. (неб.регуляризация)

Посмотрим рез-ты применения к набору д-х 2\_монс разл.знач. альфа и исп-нием 2 скрытых слоев с 10 и 1000 эл-тами в к.[9]

Т.о. сущ-ет много способов регулир.сложность нейр.сети:

* кол-во скрытых слоев
* кол-во эл-тов в к.скрытом слое
* регуляризация (alpha)

Важным св-вом нейронных сетей явл.то, что их веса задаются случ.образом перед началом обучения и случ.инициализ.влияет на процесс обучения модели.

Т.е. даже при исп-нии одних и тех же пар-ров мы м.получить очень разные модели.

При условии, что сеть им.большой размер и ее сложность настроена правильно, данный факт не д.сильно влиять на правильность, однако стоит об этом помнить при работе с неб.сетями).

Графики моделей, обученных с пом.тех же знач.пар-ров, но с разных стартовых значений.[10]

Для того, чтобы лучше понять как нейронная сеть работает на реальных данных применим классификатор к набору данных рака. Начнем с параметров по умолчанию [11].

Правильность на таком наборе [12]

Довольно неплохая правильность, но не такая хорошая как у нек.других моделей. Это обусловлено масштабом данных. Нейронные сети также как и ядерный метод опорных векторов требует, чтобы все вх.признаки были измерены в одном и том же масштабе, в идеале они должны иметь среднее 0 и дисперсию 1.

Отмасштабируем даные вручную.[13]

После масштабирования данные стали намного лучше и теперь легче конкурировать с рез-тами остальных моделей. Мы получили предупр.что достигнуто мкс.кол-во итераций. Это неотъемлимая часть алг.adam и сообщ.о том, что мы д.увел.кол-во итераций. [14]

Т.к.сущ-ет опред.разрыв между правильностью на обуч.и тест.наборах => уменьшим сложность модели, чтобы улучшить обобщ.спос.

В данном случ.увел.пар-р alpha (c 0.0001 до 1) чтобы прим.более строгую регуляриз.[15]

Получаем правильность сопоставимую с правильностью лучших моделей.

Построим график для того, чтобы показать весовые коэф., кот.выли вычислены при подключении вх.слоя к перому слою.[16]

Строки – 30 вх.признаков

Столбцы – 100 скрытых эл-тов.

Светлые цвета – выс.полож.знач.

Темные цвета – отриц.знач.

=> 1) признаки с небольшими весами скрытых эл-тов «менее важны» в модели