Домашнее задание №2 по курсу «Машинное обучение»: линейные модели

Колесов Алексей

10 сентября 2019 г.

В первой задаче вам необходимо прислать python-файл `features.py`, в комментариях которого должны быть указаны подобранные признаки и их мотивация.

Для остальных задач необходимо прислать pdf-файл.

1 Задания

1. В этой задаче вам необходимо будет произвести `feature-engineering` на примере датасета MNIST — `http://yann.lecun.com/exdb/mnist/`. Датасет представляет собой чёрно-белые изображения рукописных цифр, разделённых на десять классов: по признаку того, какая цифра была написана. На этом датасете можно решать 45 задач бинарной классификации: каждая цифра против каждой. Ваша задача для каждой из этих задач придумать по два признака, так чтоб логистическая регрессия решала бы задачу как минимум с точностью 80 процентов.

   Про логистическую регрессию необходимо знать только то, что это один из методов построить линейный классификатор для случая, когда выбора не является разделимой.

   Для решения этой задачи вам предоставлены два файла: `main.py` с кодом, который скачивает датасет, составляет 45 задач, обучает для каждой задачи логистическую регрессию и измеряет полученный результат. Этот файл вам менять не нужно, но стоит ознакомиться для понимания.

   Второй файл — `features.py` — пример которого вам тоже дан, должен быть написан вами. В нём необходимо реализовать набор функций, принимающих один аргумент — вектор размерности 784 (исходные картинки имеют размер 28×28) и выдаёт единственное вещественное число.

   Файл `features.py` должен экспортировать переменную `FEATURES` — `dict` из пары цифр в пару функций, вычисляющих признаки. Внутри этого файла нельзя использовать ничего кроме стандартной библиотеки и библиотеки `numpy`.

   Задача считается решённой, если скрипт отработал менее, чем за 20 минут и решил как минимум 40 случаев. Кроме того отдельные баллы будут засчитаны решениям, которые будут в топе по средней точности, минимальной точности (которую надо максимизировать) и количеству решённых задач.

   Запуск файла `main.py` выдаст вышеописанные числа на стандартный вывод.

2. Покажите, что задачу минимизации эмпирического риска в линейной регрессии со следующей функцией потерь $l(h, (x, y)) = |h(x) - y|$ можно представить, как задачу линейного программирования.

   Задача, о которой идёт речь:

   $$\min_w \sum_{i=1}^{m} |\langle w, x_i \rangle - y_i|$$

   Подсказка: докажите сначала, что для любого $c \in \mathbb{R}$, $|c| = \min_{a \geqslant 0} a$, такое что $c \leqslant a$ и $c \geqslant -a$

3. Допустим, что в алгоритме Perceptron вместо правила $w^{(t+1)} = w^{(t)} + y_i x_i$ мы применяем правило $w^{(t+1)} = w^{(t)} + \eta y_i x_i$ для некого $\eta > 0$ (одинакового для всех шагов). Пусть оба алгоритма выбирают объект для шага одинаковым образом. Докажите, что модифицированный алгоритм сделает такое же количество шагов и сойдётся с вектором, который сонаправлен с вектором немодифицированного алгоритма.