Школа анализа данных
Машинное обучение, часть 2
Домашнее задание №1

Решите предложенные задачи. Решения необходимо оформить в виде PDF документа. Каждая задача должна быть подробно обоснована, задачи без обоснования не засчитываются. Решения пишутся в свободной форме, однако так, чтобы проверяющие смогли разобраться. Если проверяющие не смогут разобраться в решении какой-нибудь задачи, то она автоматически не засчитывается. Дедлайн очников 17 октября 2017 09:00MSK, дедлайн заочников и филиалов +2 суток.

Задача 1 (2 балла) Композиции алгоритмов, бустинг, AdaBoost.

Обозначим через $\tilde{w}^{(N)}$ нормированный вектор весов на N-й итерации алгоритма AdaBoost. Покажите, что взвешенная ошибка базового классификатора b_N относительно весов со следующего шага $\tilde{w}_i^{(N+1)}$ равна 1/2:

$$\sum_{i=1}^{\ell} \tilde{w}_i^{(N+1)} [b_N(x_i) \neq y_i] = \frac{1}{2}.$$

Задача 2. Градиентный бустинг.

1. **(1 балл)** Какой функции потерь будет соответствовать градиентный бустинг, который на каждой итерации настраивается на разность между вектором истинных меток и текущим вектором предсказанных меток?

2. **(1 балл)** Градиентный бустинг обучается на пяти объектах с функцией потерь для одного объекта

   $$\mathcal{L}(\tilde{y}, y) = (\tilde{y} - y)^4.$$

   На некоторой итерации полученная композиция дает ответ $(5, 10, 6, 3, 0)$. На какой вектор ответов будет настраиваться следующий базовый алгоритм, если истинный вектор ответов равен $(6, 8, 6, 4, 1)$?

3. **(1 балл)** Рассмотрим задачу бинарной классификации, $= \{0, 1\}$. Будем считать, что все алгоритмы из базового семейства $\mathcal{A}$ возвращают ответы из отрезка $[0, 1]$, которые можно интерпретировать как вероятности принадлежности объекта к классу 1. В качестве функции потерь возьмем отрицательный логарифм правдоподобия (negative log-likelihood):

   $$L(y, z) = -\big(y \log z + (1 - y) \log(1 - z)\big),$$

   где y — правильный ответ, а z — ответ алгоритма.

   Выпишите формулы для поиска базовых алгоритмов b_n и коэффициентов γ_n в градиентном бустинге.

Задача 3 (2 балла) Композиции, устойчивость к шуму.

1. Рассмотрим алгоритм AdaBoost — бустинг с экспоненциальной функцией потерь

$$\mathcal{L}(M) = \exp(-M),$$

   где M — отступ объекта. Покажите, что алгоритм неустойчив к шуму, т.е. возможен неограниченный рост отношения весов шумовых объектов по отношению к весам пороговых объектов.

2. Покажите, что бустинг с логистической функцией потерь

$$\mathcal{L}(M) = \log(1 + \exp(-M))$$

   устойчив к шуму в описанном выше смысле смысле.

Примечание. Пороговые объекты — это те, для которых значение отступа положительно и порядка нуля, то есть они лежат близко к границе между классами и в своем классе. Шумовые объекты лежат глубоко в чужом классе, на них отступ принимает большие отрицательные значения.

Задача 4 (2 балла) Ранжирование.

Имеются два ранжирующих правила $h_1(x)$ и $h_2(x)$. Дана выборка из одного запроса и N документов. Релевантности документов и ответы каждого из ранжирующих правил на каждом документе этой выборки известны. Предложите точный алгоритм построения линейной композиции

$$h(x) = \alpha h_1(x) + (1 - \alpha) h_2(x), \quad \alpha \in [0, 1]$$

двух ранкеров, имеющей оптимальное значение NDCG на этой выборке. Сложность алгоритма должна быть не выше $O(N^3)$.

Задача 5 (1 балл). CRR — combined regression and ranking.

Рассмотрим задачу ранжирования рекламных объявлений для показа на интернет-странице. Пусть в качестве обучающей выборки нам дано множество показов $X = \{x_i\}_{i=1}^{l}$ различных рекламных объявлений. Каждый показ x_i описывается набором признаков и ответом $y_i \in \{-1, +1\}$, где $y_i = +1$ соответствует случаю, когда для данного показа был клик по объявлению, а $y_i = -1$ — случаю, когда клика не было. Поскольку за один раз на странице отрисовывается несколько рекламных объявлений, то показы объединены в группы $g_1, \ldots, g_m$ размером от одного до пяти ($|g_k| \in \{1, \ldots, 5\} \quad \forall k \in \{1, \ldots, m\}$) так, что показы из одной группы происходили одновременно. Показ группы объявлений можно мыслить себе как аналогию показа некольких документов в ответе на конкретный поисковый запрос. Для каждого показа известен номер группы k, в которой он произошёл. Каждый показ принадлежит только одной группе, причем в группе может быть несколько кликов, или же не быть кликов вовсе. Задача ранжирования состоит в том, чтобы итоговая модель предсказывала кликнутым объявлениям ответ больший, чем остальным объявлениям из той же группы.

Предложите функцию потерь, которая бы учитывала следующие требования: построенная модель одновременно пытается решить задачи ранжирования и предсказания вероятности клика. Теперь о том, зачем это нужно. На практике возникает необходимость, чтобы модель была не только ранжирующей, но и ответ модели был бы интерпретируем как вероятность клика по рекламному объявлению. Это нужно для того, чтобы для каждого объявления x_i считать матожидание заработанных с объявления денег. Делается это по простой формуле: $CPC_i \cdot p_i$, где CPC_i — стоимость клика (Cost Per Click), а p_i — ответ модели на i-ом примере. Именно поэтому хочется, чтобы ранжирующая модель выдавала вероятности — ранжирование нужно для того, чтобы выдавать пользователю наиболее релевантный контент, а вероятностная интерпретация нужна для оценки прибыли.