

# Машинное обучение, ФКН ВШЭ

## Семинар №23

### 1 Обучение модели попарных соотношений

Ранее в курсе мы рассматривали задачи обучения с учителем, в которых данные имеют вид  $\{(x_i, y_i)\}_{i=1}^{\ell}$ , где  $x_i$  — вектор признаков, а  $y_i$  — таргет, который необходимо предсказать по признакам (задачи классификации и регрессии). Также мы рассматривали задачи, где данные имеют вид  $\{x_i\}_{i=1}^{\ell}$  — задачи кластеризации или задачи понижения размерности.

На лекции мы рассмотрели новую постановку — задачу ранжирования и кратко рассмотрели существующие подходы (pointwise, pairwise, listwise). Несмотря на то, что во всех подходах целью является верное ранжирование объектов, считается, что заранее известно некоторое истинное значение релевантности (которое может быть основано на некоторой оценке ассессора, клике по документу или каком-либо другом сигнале), которое используется в оптимизируемом функционале качества. Однако зачастую данные в подобных задачах представлены лишь попарными взаимодействиями между объектами, то есть для некоторого набора объектов  $\{x_i\}_{i=1}^n$  обучающая выборка представлена случайным подмножеством попарных соотношений:  $\{x_{i_k} > x_{j_k}\}_{k=1}^{\ell}$ . Также для таких соотношений может быть известен некий контекст.

Далее мы будем рассматривать модели для попарных соотношений в контексте матчей между двумя игроками  $x_i, x_j$ , при этом наши рассуждения могут быть полностью перенесены на все остальные случаи применения этих моделей.

В общем случае данные о попарных соотношениях выглядят следующим образом:

- признаки объектов — для каждого игрока известно его признаковое описание  $x_i \in \mathbb{R}^d$  (id, вес, рост, национальность и т.п.);
- признаки контекста — для каждого матча известно его признаковое описание  $z_k \in \mathbb{R}^D$  (погода, время, место и т.п.);
- набор соотношений — история матчей в виде троек  $\{(x_{i_k}, x_{j_k}, z_k)\}$ , где на первом месте стоит победитель матча.

**Задача 1.1.** Придумайте признаки объектов и признаки контекста для следующих задач:

- игра в теннис,
- рекомендации ресторанов,
- идентификация личности,
- клики на поисковую выдачу.

**Решение.**

- **В игре в теннис** объектом является игрок (теннисист), а соотношением является матч между двумя игроками. Таким образом, признаками игрока могут быть: пол, возраст, национальность, рост, вес и т.д. Признаками контекста матча могут быть: тип покрытия корта, погодные условия во время матча, уровень турнира, в рамках которого состоялась встреча.
- **При рекомендации ресторанов** объектом является ресторан, а соотношением является событие, когда пользователь предпочёл один ресторан другому. Признаками ресторана могут быть: тип кухни, средний чек, месторасположение, наличие детской комнаты, оценка пользователей. В качестве признаков контекста можно использовать признаки пользователя, который выбирал между ресторанами: где он выбирал ресторан (сайт или мобильное приложение), как составил запрос, в какое время он выбирал рестораны.
- **Для идентификации личности** можно поставить следующую задачу – есть входная фотография человека, фотография из базы данных, запись в базе данных, необходимо определить относится ли запись в базе данных и фотография из базы данных к человеку на входной фотографии. В качестве объекта мы выбираем фотографию человека, а в качестве контекста запись в базе данных. Признаковое описание фотографии можно получить, например, с помощью нейронных сетей, а признаковое описание объекта можно составить из записи в базе данных.
- Пусть у нас есть задача попарного сравнения документов в контексте одного запроса. Выборку набираем следующим образом – документ, на который кликнул человек в поисковой выдаче лучше всех документов, которые стоят выше него и на которые человек не кликнул. Признаки объекта это признаки документа: bm25, ctr, порядковый номер в выдаче и т.д., признаки контекста – длина запроса, размер выдачи на странице, версия поисковика (мобильная, браузерная) и т.д.

■

## §1.1 Rote learning (зазубривание)

Самая простая модель, которую можно обучить на этих данных это запоминание матрицы вероятностей для каждой пары игроков  $(a, b)$ . Искомую вероятность вычисляем по следующей формуле:

$$P(x_i \text{ победит } x_j) = \frac{\sum_{k=1}^{\ell} [x_{i_k} = x_i][x_{j_k} = x_j]}{\sum_{k=1}^{\ell} ([x_{i_k} = x_i][x_{j_k} = x_j] + [x_{i_k} = x_j][x_{j_k} = x_i])}. \quad (1.1)$$

Так как могут быть ситуации, когда игроки никогда не встречались до этого (то есть в числителе и знаменателе стоят нули), то необходимо произвести сглаживание формулы (1.1):

$$P(x_i \text{ победит } x_j) = \frac{\sum_{k=1}^{\ell} [x_{i_k} = x_i][x_{j_k} = x_j] + 1}{\sum_{k=1}^{\ell} ([x_{i_k} = x_i][x_{j_k} = x_j] + [x_{i_k} = x_j][x_{j_k} = x_i]) + 2}.$$

**Минусы:**

- Необходимо очень много данных.
- Не учитывает никаких признаков.
- Не даёт оценки на исход матча между игроками, которые не встречались прежде.

## §1.2 Модель Брэдли-Терри

Следующая модель, которую мы рассмотрим, сопоставляет каждому игроку  $x_i$  некоторый параметр  $\gamma_i$ , который может быть интерпретирован как сила (уровень) игрока. Вероятность победы игрока  $x_i$  над игроком  $x_j$ , в такой модели записывается как:

$$\begin{aligned} P(x_i \text{ победит } x_j) &= \frac{\exp(\gamma_i)}{\exp(\gamma_i) + \exp(\gamma_j)} \\ &= \frac{1}{1 + \exp(\gamma_j - \gamma_i)} = \sigma(M(x_i, x_j)), \end{aligned}$$

где  $\sigma(x)$  – сигма-функция, а функция  $M(x_i, x_j) = \gamma_i - \gamma_j$  – функция матча (matchup function). Функцию матча  $M(x_i, x_j)$  можно задать произвольным образом, но при этом должны выполняться условия:

1.  $M(x_i, x_j) \in \mathbb{R}$ , при этом положительные значения означают, что с большей вероятностью победит  $i$ , а отрицательные – наоборот. Вероятность победы любого из игроков при  $M(x_i, x_j) = 0$  равняется 0.5;
2. при  $M(x_i, x_j) \rightarrow +\infty$ ,  $P(x_i \text{ победит } x_j) \rightarrow 1$ , и наоборот;

3.  $M(x_i, x_j) = -M(x_j, x_i)$  — данное условие необходимо, чтобы выполнялось  $P(x_i \text{ победит } x_j) = 1 - P(x_j \text{ победит } x_i)$ .

**Плюсы:**

- Обладает небольшим числом параметров;
- Даёт оценки на исход матча между игроками, которые не встречались прежде.

**Минусы:**

- По-прежнему не учитывает никаких признаков.

**Задача 1.2.** Запишите оптимизационную задачу, которую необходимо решить для настройки параметров этой модели. Рассчитайте градиенты полученного функционала.

**Решение.**

$$\begin{aligned} \sum_{k=1}^{\ell} \log P(x_{i_k} \text{ победит } x_{j_k}) &\rightarrow \max_{\{\gamma_i\}} \\ \frac{\partial}{\partial \gamma_s} \sum_{k=1}^{\ell} \log P(x_{i_k} \text{ победит } x_{j_k}) &= \frac{\partial}{\partial \gamma_s} \sum_{k=1}^{\ell} [i_k = s] \log P(x_{i_k} \text{ победит } x_{j_k}) \\ &\quad + \frac{\partial}{\partial \gamma_s} \sum_{k=1}^{\ell} [j_k = s] \log P(x_{i_k} \text{ победит } x_{j_k}) \\ &= \sum_{k=1}^{\ell} [i_k = s] \left( \frac{1}{\sigma(M(s, x_{j_k}))} \sigma(M(s, x_{j_k}))(1 - \sigma(M(s, x_{j_k}))) \right) \\ &\quad + \sum_{k=1}^{\ell} -[j_k = s] \left( \frac{1}{\sigma(M(x_{i_k}, s))} \sigma(M(x_{i_k}, s))(1 - \sigma(M(x_{i_k}, s))) \right) \\ &= \sum_{k=1}^{\ell} [i_k = s] \left( \frac{\exp(\gamma_{j_k})}{\exp(\gamma_s) + \exp(\gamma_{j_k})} \right) - \sum_{k=1}^{\ell} [j_k = s] \left( \frac{\exp(\gamma_s)}{\exp(\gamma_{i_k}) + \exp(\gamma_s)} \right) \end{aligned}$$

■

### §1.3 Pairwise logistic regression model

Модель Брэдли-Терри может быть легко расширена для учёта признаков игроков, если вместо в качестве параметров рассматривать не  $\gamma_i$ , а обучить линейную модель для оценки силы игрока, то есть сила  $\gamma_i$  игрока  $x_i$  будет выражаться следующим образом:

$$\gamma_i = w^T x_i.$$

Заметим, что если в качестве признакового описания игрока мы возьмём его id, представленный с помощью one-hot-encode вектора, то получим модель Брэдли-Терри в явном виде.

При обучении этой модели можно добавлять регуляризацию на веса  $w$ .

Тем не менее, модель по-прежнему не может учесть признаки матча. Действительно, попробуем добавить признаки матча  $z_g$  к признаковому вектору каждого игрока:

$$M(a, b) = w^T \left( \begin{bmatrix} x_i \\ z_k \end{bmatrix} - \begin{bmatrix} x_j \\ z_k \end{bmatrix} \right) = w^T \begin{bmatrix} x_i - x_j \\ 0 \end{bmatrix}.$$

Видим, что признаковое описание матча не влияет на функцию  $M(x_i, x_j)$ , а потому не участвует в оптимизируемом функционале. Попробуем переписать функцию  $M(x_i, x_j)$  так, чтобы она явно учитывала признаки матча линейным образом:

$$M(x_i, x_j) = w^T(x_i - x_j) + w_k^T z_k,$$

но заметим, что в этом случае не выполняется условие 3 для функции матча  $M(x_i, x_j)$ .

## §1.4 Blade-chest model

Следующий подход, который мы рассмотрим, предлагает оригинальный вариант функции  $M(a, b)$ . Идея подхода заключается в том, что представление силы игрока одним числом плохо описывает данные, в частности, такая модель не может описать соотношение "камень-ножницы-бумага" между тремя игроками. Поэтому предлагается обучать для каждого игрока  $x_i$  два вектора —  $x_i^{\text{blade}}$  и  $x_i^{\text{chest}}$ . Чтобы оценить вероятность победы игрока  $x_i$  над игроком  $x_j$  необходимо сравнить расстояния между представлениями — если  $x_i^{\text{blade}}$  ближе к  $x_j^{\text{chest}}$ , чем  $x_j^{\text{blade}}$  ближе к  $x_i^{\text{chest}}$ , то вероятнее победит игрок  $x_i$ . Названия представлений (клинок и грудь) дают понятную интерпретацию этих векторов. На рис. 1 эта интерпретация проиллюстрирована.

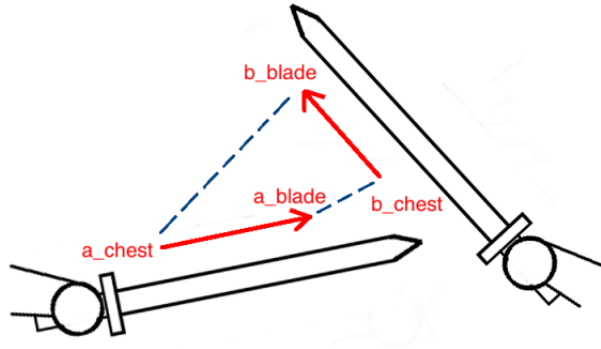


Рис. 1. На рисунке проиллюстрирована интерпретация обучаемых представлений. Между игроками  $a$  и  $b$  происходит поединок. Видно, что клинок игрока  $a$  ближе к груди игрока  $b$ , чем клинок игрока  $b$  к груди игрока  $a$ , следовательно игрок  $a$  имеет более высокие шансы на победу.

Функция  $M(x_i, x_j)$  в этой модели записывается как

$$M(x_i, x_j) = \|x_j^{\text{blade}} - x_i^{\text{chest}}\|^2 - \|x_i^{\text{blade}} - x_j^{\text{chest}}\|^2.$$

Также можно записать функцию  $M(x_i, x_j)$  альтернативным способом, используя скалярное произведение:

$$M(x_i, x_j) = \langle x_i^{\text{blade}}, x_j^{\text{chest}} \rangle - \langle x_j^{\text{blade}}, x_i^{\text{chest}} \rangle.$$

В предыдущей модели мы использовали линейную комбинацию признаков для оценки силы игрока. Тот же подход мы можем использовать для оценки искомых представлений. Кроме того, помимо линейной комбинации признаков игроков, можно также использовать любую дифференцируемую функцию для оценки этих векторов. В общем виде можем записать:

$$\begin{aligned} x_i^{\text{blade}} &= f_{\text{blade}}(Bx_i) \\ x_j^{\text{blade}} &= f_{\text{blade}}(Bx_j) \\ x_i^{\text{chest}} &= f_{\text{chest}}(Cx_i) \\ x_j^{\text{chest}} &= f_{\text{chest}}(Cx_j). \end{aligned}$$

Получение данных представлений можно рассматривать как полносвязный слой нейросети с нелинейностью. Без ограничений общности мы можем использовать любые архитектуры нейросетей для обучения данных представлений.

В данной модели мы можем учесть признаки игры  $z_k$  одним из двух способов:

1. Первый способ заключается в добавлении признаков игры к признаковому описанию каждого игрока:

$$\begin{aligned} x_i^{\text{blade}} &= f_{\text{blade}}\left(B \begin{bmatrix} x_i \\ z_k \end{bmatrix}\right), \\ x_i^{\text{chest}} &= f_{\text{chest}}\left(C \begin{bmatrix} x_i \\ z_k \end{bmatrix}\right). \end{aligned}$$

2. Второй способ заключается в **поэлементном** домножении векторов представлений на векторы представлений матча, полученные из признакового описания матча аналогично векторам  $x^{\text{blade}}$ ,  $x^{\text{chest}}$ , то есть:

$$\begin{aligned} x_i^{\text{blade}} &= f_{\text{blade}}(Bx_i) \cdot f_{\text{match}}(B'z_k), \\ x_i^{\text{chest}} &= f_{\text{chest}}(Cx_i) \cdot f_{\text{match}}(C'z_k). \end{aligned}$$