Машинное обучение, ФКН ВШЭ
Семинар №20

1 Вывод ALS и HALS

На лекции была поставлена задача построения модели со скрытыми переменными (latent factor model) для коллаборативной фильтрации:

$$\sum_{u,i}(r_{ui} - \langle p_u, q_i\rangle)^2 \to \min_{P,Q},$$

напомним, что суммирование ведется по всем u, i для которых известен рейтинг – r_{ui}, а p_u, q_i – латентные представления пользователя u и товара i, соответственно.

§1.1 Alternating Least Squares

Alternating Least Squares подход решает задачу (1) попеременно фиксируя матрицы P и Q, оказывается, что зафиксировав одну из матриц легко можно выписать аналитическое решение задачи (1) для другой.

$$\nabla_{p_u}\left[\sum_{u,i}(r_{ui} - \langle p_u, q_i\rangle)^2\right] = \sum_i 2(r_{ui} - \langle p_u, q_i\rangle)q_i = 0$$

Воспользовавшись тем, что $a^T bc = cb^T a$, получим

$$\sum_i r_{ui}q_i - \sum_i q_i q_i^T p_u = 0.$$

Тогда окончательно каждый столбец матрицы P можно найти по формуле

$$p_u = \left(\sum_i q_i q_i^T\right)^{-1} \sum_i r_{ui}q_i \quad \forall u,$$

аналогично для столбцов матрицы Q

$$q_i = \left(\sum_u p_u p_u^T\right)^{-1} \sum_u r_{ui}p_u \quad \forall i.$$

Таким образом мы можем решать оптимизационную задачу (1) поочередно фиксируя одну из матриц P или Q и проводя оптимизацию по второй.

§1.2 Hierarchical Alternating Least Squares

Подход ALS требует вычислений обратных матриц, что накладывает сильные ограничения на размерность скрытых переменных. Альтернативой ALS является подход HALS (Hierarchical Alternating Least Squares), в котором на каждой итерации мы решаем оптимизационную задачу (1) относительно только одной строчки матрицы P или Q. Выведем аналитические формулы для k-ой строчки матрицы P.

$$\frac{\partial}{\partial P_{ku}}\left[\sum_{u,i}(r_{ui} - \langle p_u, q_i\rangle)^2\right] = \sum_i 2(r_{ui} - \langle p_u, q_i\rangle)Q_{ki} = 0$$

$$\sum_i (r_{ui} - \sum_{s\neq k} P_{su}Q_{si})Q_{ki} - P_{ku}\sum_i Q_{ki}^2 = 0$$

$$P_{ku} = \frac{\sum_i (r_{ui} - \sum_{s\neq k} P_{su}Q_{si})Q_{ki}}{\sum_i Q_{ki}^2}$$

Если обозначить p^k как строчку матрицы P, то в векторном виде формула запишется следующим образом:

$$p^k = \frac{(R - \sum_{s\neq k} p^s q^{sT})q^k}{q^{kT}q^k}.$$

Аналогично для строчки матрицы Q,

$$q^k = \frac{(R - \sum_{s\neq k} p^s q^{sT})^T p^k}{p^{kT}p^k}.$$

В этих формулах, не смотря на то, что p^k – строчка матрицы P, мы оперируем ей как вектор-столбцом. Иначе говоря, p^k – транспонированная k-ая строчка матрицы P.

2 Neural Colloborative Filtering

Нелинейным обобщением матричных разложений является коллаборативная фильтрация с помощью нейронных сетей. Сопоставим каждому пользователю u one-hot encode вектор α_u, у которого на u-ом месте стоит 1, а остальные координаты заполнены нулями. Аналогичный вектор β_i определим для товара i. В качестве алгоритма классификации (или регрессии, в зависимости от постановки) будем использовать нейросеть, у которой два полносвязных слоя на входе – один соответствует пользователям и принимает вектор α_u, второй соответствует товарам и принимает на вход вектор β_i. После входных полносвязных слоев, соответствующие представления необходимо сконкатенировать и передать в следующие полносвязные слои (см. рис. 1). На выходе нейронной сети мы получаем пресказание $\widehat{y}_{ui}$, которое сравниваем с истинным ответом для заданной пары r_{ui} (на картинке y_{ui}).

Не смотря на свою простоту модель обладает рядом важных достоинств:

- т.к. обучать такую модель мы можем только с помощью стохастического градиентного спуска, мы можем использовать различные вариации функции потерь, например, использовать log-loss для задачи классификации;

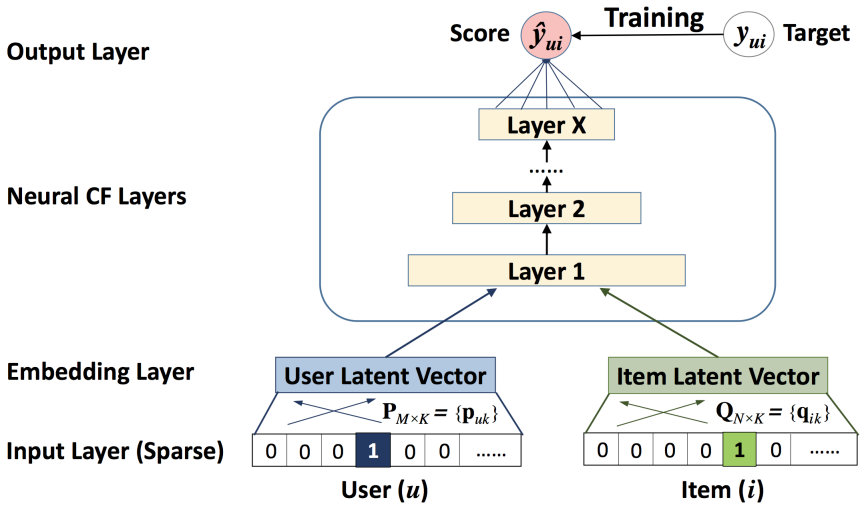

Рис. 1. Neural Collaborative Filtering

- мы можем легко обобщить модель на случай, когда у нас есть признаки пользователей или товаров, добавив эти признаки к векторам α_u и β_i соответственно (также необходимо заметить, что добавление признаков в модель позволяет частично решить проблему холодного старта);

- в зависимости от размеров выборки и природы данных мы можем адаптировать архитектуру под свои нужды, например, если у нас есть картинки в качестве признаков товаров, мы можем добавить сверточные слои в нашу нейронную сеть и обучать всю модель end-to-end.

В этой модели мы по прежнему можем получить латентные представления пользователей и товаров. На самом деле, латентными представлениями будут столбцы матрицы весов в первых полносвязных слоях (в предположении, что мы умножаем матрицу весов на вектор признаков справа).

3 Факторизационные машины

Пусть обучающая выборка представлена матрицей $X \in \mathbb{R}^{\ell \times d}$, где ℓ – кол-во объектов, а d – кол-во признаков. В случае решения задачи построения рекомендательной системы объектами, как правило, являются пары пользователь-товар, для которых известны оценки r_{ui} степени заинтересованности пользователя в товаре. При этом признаки могут разбиваться на группы, описывающие различные составляющие информации о паре (см рис. 2).

Факторизационные машины учитывают взаимодействия признаков вплоть до некоторой степени, которая чаще всего равна двум. В этом случае модель будет выглядеть следующим образом:

$$a(x) = w_0 + \sum_{j=1}^{d} w_j x_j + \frac{1}{2} \sum_{j=1}^{d} \sum_{k=1, k\neq j}^{d} \langle v_j, v_k \rangle x_j x_k, \tag{3.1}$$

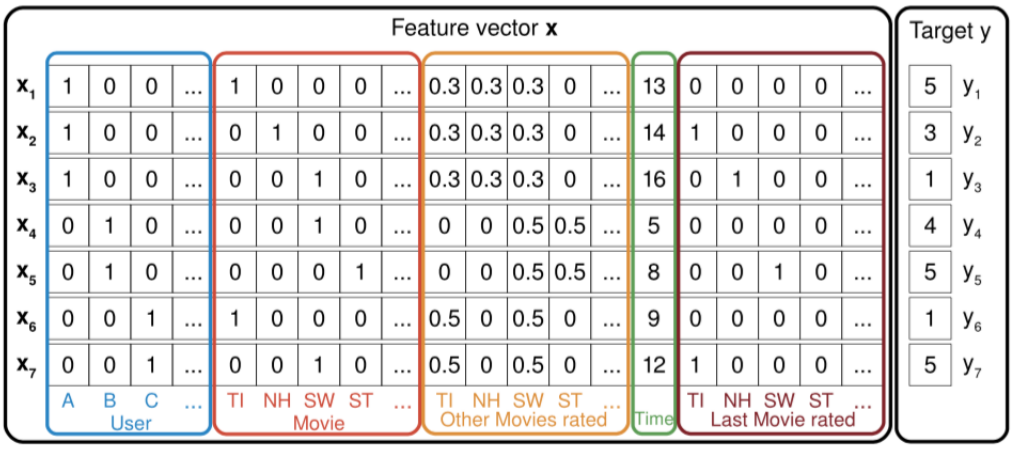

Рис. 2. Представление даннных

где $w_0, w_1, \ldots, w_d \in \mathbb{R}$, $v_1, \ldots, v_d \in \mathbb{R}^r$ — параметры модели, $x = (x_1, \ldots, x_d)$ — признаковое описание обхекта. Первое слагаемое отвечает константному предсказанию, второе слагаемое является линейной моделью, третье слагаемое содержит все попарные взаимодействия $x_j x_k$. В отличие от полиномиальной модели, где каждому попарному взаимодействию x_j, x_k ставится индивидуальный вес w_{jk}, здесь вес взаимодействия моделируется скалярным произведением двух векторов $\langle v_j, v_k \rangle$. Иначе можно сказать, что матрица всех весов при слагаемых второго порядка W представима как $W = VV^T$, где $V = [v_1, \ldots, v_d]^T$.

Из разложения $W = VV^T$ можно сделать выводы о том, в каких ситуациях применяются факторизационные машины — в частности, в случаях, когда количество признаков d очень велико. При $d >> \ell$ обучение полной матрицы весов попарных взаимодействий быстро приводит к переобучению, поэтому в подобных ситуациях параметр r обычно выбирают много меньше d.

§3.1 Методы обучения

Пусть имеется обучающая выборка $X = \{(x_i,\ y_i)\}_{i=1}^{\ell}$. Параметрами модели являются $\Theta = (w_1, \ldots, w_d, \mathbf{v_1}, \ldots, \mathbf{v_d}) = (w_1, \ldots, w_d, v_{11}, \ldots, v_{dr})$. Задача обучения факторизационной машины формулируется стандартным образом:

$$Q(a, X) = \sum_{i=1}^{\ell} L(a(x_i; \Theta), y_i) + \sum_{\theta \in \Theta} \lambda_\theta \theta^2 \to \min_{\Theta}, \tag{3.2}$$

где λ_θ — коэффициент регуляризации параметра θ, $L(z, y)$ — функция потерь, которая в случае решения задачи регрессии может задаваться как MSE:

$$L(z, y) = (z - y)^2,$$

а в случае бинарной классификации (для $y \in \{0, 1\}$):

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

Второе слагаемое является регуляризатором и может индивидуально выбираться для каждого θ. На практике параметры разбивают на некоторые группы (как на рис. 2), для каждой из которых выбирают индивидуальный параметр λ_θ.

Существует несколько способов обучения факторизационных машин:

- SGD (stochastic gradient descent) — оптимизация параметров происходит методом стохастического градиентного спуска по объектам обучающей выборки.
- ALS (Alternating Least-Squares) — оптимизация параметров происходит попеременно по каждому параметру при фиксированных значениях остальных параметров.
- MCMC (Markov Chain Monte Carlo) — параметры сэмплируются из апостериорного распределения.

3.1.1 SGD

Обучение факторизационных машин при помощи SGD осуществуется стандартным образом — случайным образом выбирается один из объектов обучающей выборки, после чего обновляются все параметры модели. Ниже приведён подробный алгоритм обучения.

Вход: Обучающая выборка $X = \{(x_i, y_i)\}_{i=1}^{\ell}$, параметры регуляризации $\{\lambda_\theta\}_{\theta\in\Theta}$, шаг градиентного спуска η, дисперсия инициализации σ.
Выход: Параметры модели $\Theta^* = (w_0^*, \ldots, w_d^*, \mathbf{v_1}^*, \ldots, \mathbf{v_d}^*)$.

$w_j := 0,\ j = \overline{0,d}, v_{js} \sim \mathcal{N}(0,\sigma), j = \overline{1,d},\ s = \overline{1,r}$;
повторять
  для $(x_i,\ y_i) \in S$
    $w_0 := w_0 - \eta\left(\frac{\partial}{\partial w_0} L(a(x_i;\Theta), y_i) + 2\lambda_0 w_0\right)$
    для $j \in \{1,\ldots,d\} \wedge x_{ij} \neq 0$
      $w_j := w_j - \eta\left(\frac{\partial}{\partial w_j} L(a(x_i;\Theta), y_i) + 2\lambda_{w_j} w_j\right)$
      для $s \in \{1,\ldots,r\}$
        $v_{js} := v_{js} - \eta\left(\frac{\partial}{\partial v_{js}} L(a(x_i;\Theta), y_i) + 2\lambda_{v_{js}} v_{js}\right)$
пока веса не стабилизируются

3.1.2 ALS

Alternating Least-Squares или Coordinate Descent (координатный спуск) — метод обучения, при котором каждый параметр настраивается независимо, т.е. все параметры модели настраиваются поочередно при фиксированных значениях остальных параметров. При этом для каждого параметра можно вывести аналитическую формулу для его оптимального значения на каждом шаге данного метода обучения.

Задача 3.1. *Выразите оптимальное значение некоторого параметра $\theta \in \Theta$ при фиксированных значениях остальных параметров $\Theta\backslash\{\theta\}$ при оптимизации функционала $Q(a, X)$ для задачи регрессии с функционалом MSE.*

Решение. Заметим, что для любого параметра $\theta \in \{w_0, w_1, \ldots, w_d, v_{11}, \ldots, v_{dr}\}$ модель $a(x)$ линейна по нему и может быть линеаризована следующим образом:

$$
\begin{aligned}
&a(x) = g_\theta(x) + \theta h_\theta(x),\\
&h_{w_0}(x) = 1,\\
&h_{w_j}(x) = x_j,\ j = \overline{1,d},\\
&h_{v_{js}}(x) = x_j \sum_{k \neq j} v_{ks} x_k,\ j = \overline{1,d},\ s = \overline{1,r},
\end{aligned}
$$

где функции $g_\theta(x)$, $h_\theta(x)$ не зависят от значения параметра θ.

Запишем оптимальное значение θ^* параметра θ для решаемой оптимизационной задачи:

$$
\begin{aligned}
\theta^* &= \arg\min_\theta \left(\sum_{i=1}^{\ell} (a(x_i;\theta) - y_i)^2 + \sum_{\alpha \in \Theta} \lambda_\alpha \alpha^2 \right)\\
&= \arg\min_\theta \left(\sum_{i=1}^{\ell} (g_\theta(x_i) + \theta h_\theta(x_i) - y_i)^2 + \sum_{\alpha \in \Theta} \lambda_\alpha \alpha^2 \right)\\
&= \frac{\sum_{i=1}^{\ell} (y_i - g_\theta(x_i)) h_\theta(x_i)}{\sum_{i=1}^{\ell} h_\theta^2(x_i) + \lambda_\theta}.
\end{aligned}
$$

■

Как правило, $g_\theta(x)$ не выражают в явном виде, а используют представление $g_\theta(x) = a(x;\Theta) - \theta h_\theta(x)$, из которого можно вывести формулу обновления параметра θ при использовании метода ALS:

$$
\theta^* = \frac{\sum_{i=1}^{\ell} (y_i - g_\theta(x_i)) h_\theta(x_i)}{\sum_{i=1}^{\ell} h_\theta^2(x_i) + \lambda_\theta} = \frac{\sum_{i=1}^{\ell} (y_i - a(x_i;\Theta) + a(x_i;\Theta) - g_\theta(x_i)) h_\theta(x_i)}{\sum_{i=1}^{\ell} h_\theta^2(x_i) + \lambda_\theta} =
$$
$$
= \frac{\theta \sum_{i=1}^{\ell} h_\theta^2(x_i) + \sum_{i=1}^{\ell} h_\theta(x_i)(y_i - a(x_i;\Theta))}{\sum_{i=1}^{\ell} h_\theta^2(x_i) + \lambda_\theta}.
$$

Это выражение использует текущие значения параметров модели и её прогнозы на объектах обучающей выборки и используется для итеративного обновления каждого параметра θ при фиксированных значениях остальных параметров при использовании метода ALS для обучения модели.

§3.2 Matrix Factorization

Пусть решается задача построения рекомендательной системы, имеются множество пользователей U и множество товаров I. Воспользуемся факторизационными машинами для решения этой задачи. Естественный путь описания пары $(u, i) \in U \times I$ — это бинарный вектор, состоящий из нулей и содержащий ровно две единицы:

$$
(u, i) = (\underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|U|}, \underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|I|}),
$$

где первая единица соответствует индексу пользователя и стоит на u-ом месте, а вторая единица соответствует индексу товара и стоит на $(|U| + i)$-ом месте. Тогда

факторизационная машина, обученная на данных, представленных таким образом, будет выглядеть следующим образом:

$$a(x) = w_0 + w_u + w_{|U|+i} + \langle v_u, v_{|U|+i} \rangle. \tag{3.3}$$

Как видим, данная модель совпадает с моделью в подходе, использующем матричное разложение (со сдвигами w_u, w_i). Вектора $v_u, v_{|U|+i} \in \mathbb{R}^r$ – латентные ветора пользователей и товаров соответственно.

§3.3 Pairwise Interaction Tensor Factorization

После того, как мы выпустили первую версию рекомендательной системы, разработчики добавили в интерфейс системы проставление тегов. Теперь, помимо множества пользователей U и множества товаров I у нас есть множество тегов T. Естественный путь описания тройки $(u, i, t) \in U \times I$ — бинарный вектор, состоящий из нулей и содержащий ровно три единицы:

$$(u, i, t) = (\underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|\mathrm{U}|}, \underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|\mathrm{I}|}, \underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|\mathrm{T}|}),$$

где первая единица соответствует индексу пользователя и стоит на u-ом месте, вторая единица соответствует индексу товара и стоит на $(|U| + i)$-ом месте, а третья единица соответствует индексу тега и стоит на $(|U| + |I| + t)$-ом месте. Тогда факторизационная машина, обученная на данных, представленных таким образом, будет выглядеть следующим образом:

$$a(x) = w_0 + w_u + w_{|U|+i} + w_{|U|+|I|+t} + \langle v_u, v_{|U|+i} \rangle + \langle v_u, v_{|U|+|I|+t} \rangle + \langle v_{|U|+i}, v_{|U|+|I|+t} \rangle. \tag{3.4}$$

Как видим, данная модель совпадает с моделью Pairwise Interaction Tensor Factorization. Разумеется, пример с тегом является очень частным, и вместо тега мы можем использовать любой категориальный признак.

§3.4 SVD++

Продолжаем развитие нашей рекомендательной системы. Пусть на этот раз мы столкнулись с категориальным признаком, который принимает ненулевые значения на некотором **подмножестве** категорий. Таким образом, объект обучающей выборки представлен двумя категориальными признаками: пользователь u, товар i, и множественно-категориальным признаком l, который может принимать m ненулевых значений. На Рис. 2 первые три группы признаков соответствуют такому представлению. Как и раньше, опишем такое представление числовым вектором:

$$(u, i, l = \{l_1, \ldots l_m\}) = (\underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|\mathrm{U}|}, \underbrace{0, \ldots, 0, 1, 0, \ldots, 0}_{|\mathrm{I}|}, \underbrace{0, \ldots, 1/m, \ldots, 1/m, \ldots, 0}_{|\mathrm{L}|}),$$

Как и в предыдущих пунктах, ненулевые значения полученного вектора являются индикаторами категорий признаков (подмножества категорий в случае

множественно-категориального признака). Модель для факторизационных машин выглядит следующим образом:

$$\begin{aligned} a(x) =& w_0 + w_u + w_i + \langle v_u, v_i \rangle + \frac{1}{m} \sum_{f=1}^{k} \langle v_i, v_{l_j} \rangle \\ &+ \frac{1}{m} \sum_{j=1}^{m} w_{l_j} + \frac{1}{m} \sum_{j=1}^{m} \langle v_u, v_{l_j} \rangle + \frac{1}{m^2} \sum_{j=1}^{m} \sum_{j'}^{m} \langle v_{l_j}, v_{l'_j} \rangle \end{aligned}$$

Первая часть (первая строка уравнения) соответствует модели под названием SVD++. Вторая строчка модели содержит дополнительные взаимодействия между признаками.