

# Кодирование категориальных признаков

Кантонистова Е.О.

ВШЭ, 2021

# КОДИРОВАНИЕ КАТЕГОРИАЛЬНЫХ ПРИЗНАКОВ: ONE-HOT ENCODING

- Предположим, категориальный признак  $f_j(x)$  принимает  $t$  различных значений:  $C_1, C_2, \dots, C_t$ .

Пример: еда может быть *горькой, сладкой, солёной или кислой* (4 возможных значения признака).

# КОДИРОВАНИЕ КАТЕГОРИАЛЬНЫХ ПРИЗНАКОВ: ONE-HOT ENCODING

- Предположим, категориальный признак  $f_j(x)$  принимает  $t$  различных значений:  $C_1, C_2, \dots, C_t$ .

Пример: еда может быть *горькой, сладкой, солёной или кислой* (4 возможных значения признака).

- Заменяем категориальный признак на  $t$  бинарных признаков:  $b_i(x) = [f_j(x) = C_i]$  (индикатор события).

Тогда One-Hot кодировка для нашего примера будет следующей:

*горький* = (1,0,0,0), *сладкий* = (0,1,0,0),

*солёный* = (0,0,1,0), *кислый* = (0,0,0,1).

# СЧЁТЧИКИ

**Счётчик** (*mean target encoding*) — это вероятность получить значение целевой переменной для данного значения категориального признака.

# СЧЁТЧИКИ (ПРИМЕР)

	feature	target
<b>0</b>	Moscow	0
<b>1</b>	Moscow	1
<b>2</b>	Moscow	1
<b>3</b>	Moscow	0
<b>4</b>	Moscow	0
<b>5</b>	Tver	1
<b>6</b>	Tver	1
<b>7</b>	Tver	1
<b>8</b>	Tver	0
<b>9</b>	Klin	0
<b>10</b>	Klin	0
<b>11</b>	Tver	1

# СЧЁТЧИКИ (ПРИМЕР)

	feature	target
0	Moscow	0
1	Moscow	1
2	Moscow	1
3	Moscow	0
4	Moscow	0
5	Tver	1
6	Tver	1
7	Tver	1
8	Tver	0
9	Klin	0
10	Klin	0
11	Tver	1



	feature	feature_mean	target
0	Moscow	0.4	0
1	Moscow	0.4	1
2	Moscow	0.4	1
3	Moscow	0.4	0
4	Moscow	0.4	0
5	Tver	0.8	1
6	Tver	0.8	1
7	Tver	0.8	1
8	Tver	0.8	0
9	Klin	0.0	0
10	Klin	0.0	0
11	Tver	0.8	1

# СЧЁТЧИКИ: ПРИМЕР

city	target	0	1	2
Moscow	1	$1/4$	$1/2$	$1/4$
London	0	$1/2$	0	$1/2$
London	2	$1/2$	0	$1/2$
Kiev	1	$1/2$	$1/2$	0
Moscow	1	$1/4$	$1/2$	$1/4$
Moscow	0	$1/4$	$1/2$	$1/4$
Kiev	0	$1/2$	$1/2$	0
Moscow	2	$1/4$	$1/2$	$1/4$

# СЧЁТЧИКИ В ЗАДАЧЕ БИНАРНОЙ КЛАССИФИКАЦИИ

В случае бинарной классификации счётчики можно задать формулой:

$$Likelihood = \frac{Goods}{Goods + Bads} = mean(target),$$

где *Goods* – число единиц в столбце *target*,

*Bads* – число нулей в столбце *target*.



# СЧЁТЧИКИ (ОБЩАЯ ФОРМУЛА)

- Пусть целевая переменная  $y$  принимает значения от 1 до  $K$ .
- Закодируем категориальную переменную  $f(x)$  следующим способом:

$$counts(u, X) = \sum_{(x,y) \in X} [f(x) = u]$$

$$successes_k(u, X) = \sum_{(x,y) \in X} [f(x) = u][y = k], k = 1, \dots, K$$

Тогда кодировка:

$$mean\_target_k(x, X) = \frac{successes_k(f(x), X)}{counts(f(x), X)} \approx p(y = k | f(x))$$

# СЧЁТЧИКИ (ОБЩАЯ ФОРМУЛА)

$$counts(u, X) = \sum_{(x,y) \in X} [f(x) = u]$$

$$successes_k(u, X) = \sum_{(x,y) \in X} [f(x) = u][y = k], k = 1, \dots, K$$

Тогда кодировка:

$$mean\_target_k(x, X) = \frac{successes_k(f(x), X)}{counts(f(x), X)}$$

*Недостаток? Когда такой способ кодирования переобучит наш алгоритм?*

# СЧЁТЧИКИ (ОБЩАЯ ФОРМУЛА)

$$counts(u, X) = \sum_{(x,y) \in X} [f(x) = u]$$

$$successes_k(u, X) = \sum_{(x,y) \in X} [f(x) = u][y = k], k = 1, \dots, K$$

Тогда кодировка:

$$mean\_target_k(x, X) = \frac{successes_k(f(x), X)}{counts(f(x), X)}$$

*Недостаток? Когда такой способ кодирования переобучит наш алгоритм?*

*Ответ: если в данных много редких категорий.*

# СЧЁТЧИКИ: ОПАСНОСТЬ ПЕРЕОБУЧЕНИЯ

*Вычисляя счётчики, мы закладываем в признаки информацию о целевой переменной  $y$ , тем самым, переобучаемся!*

# РЕШЕНИЕ 1: СГЛАЖИВАНИЕ

Используем счётчики (mean target encoding) со сглаживанием:

$$\frac{\textit{mean}(\textit{target}) \cdot n_{\textit{rows}} + \textit{global mean} \cdot \alpha}{n_{\textit{rows}} + \alpha},$$

$n_{\textit{rows}}$  - количество строк в категории,

$\alpha$  – параметр регуляризации.

# РЕШЕНИЕ 2: TRAIN-TEST SPLIT

- Можно вычислять счётчики так:

city	target	
Moscow	1	Вычисляем счетчики по этой части
London	0	
London	2	
Kiev	1	
Moscow	1	Кодируем признак вычисленными счётчиками и обучаемся по этой части
Moscow	0	
Kiev	0	
Moscow	2	

## РЕШЕНИЕ 2\*: КРОСС-ВАЛИДАЦИЯ

Более продвинутый способ (по кросс-валидации):

1) Разбиваем выборку

на  $m$  частей  $X_1, \dots, X_m$

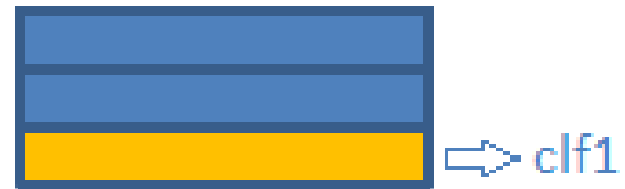
2) На каждой части  $X_i$

значения признаков

вычисляются по

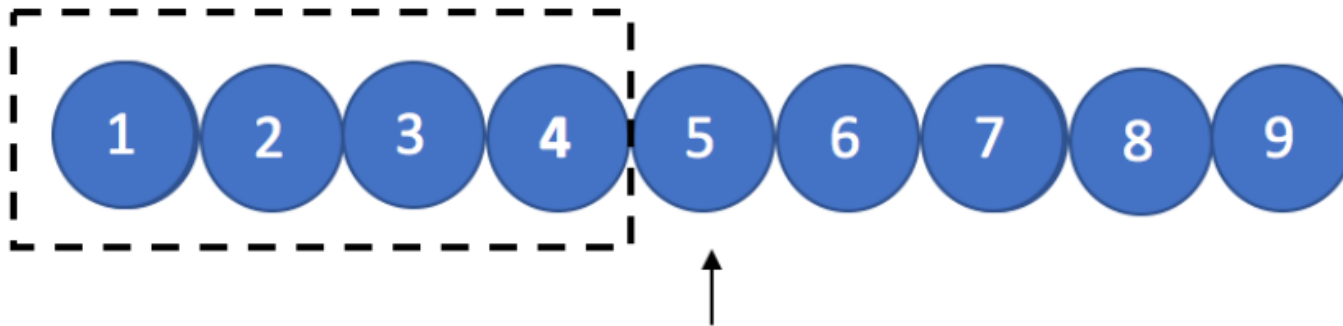
оставшимся частям:

$$x \in X_i \Rightarrow g_k(x) = g_k(x, X \setminus X_i)$$



# РЕШЕНИЕ 3: EXPANDING MEAN-SCHEMA

Суть схемы заключается в том, чтобы пройти по отсортированному в определенном порядке датасету и для подсчета счетчика для строки  $m$  использовать строки от 0 до  $m-1$ .



Running mean calculation.

Numbers are assigned randomly to each observation. Only 1-4 are used to find encoding for 5



# БОРЬБА С ПЕРЕОБУЧЕНИЕМ В СЧЁТЧИКАХ

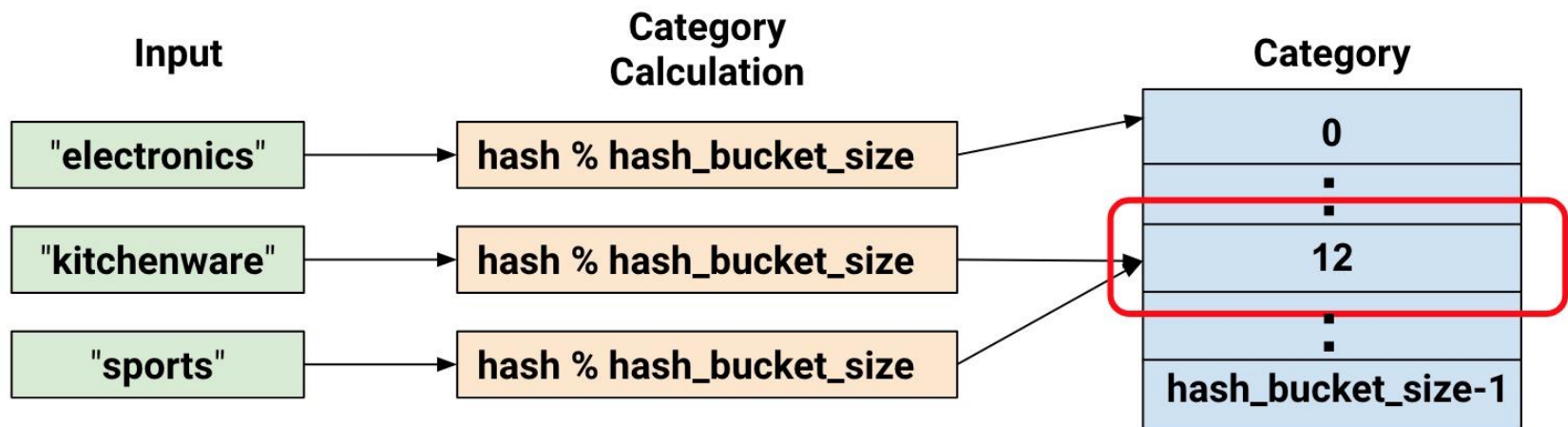
- Вычисление счётчиков по кросс-валидации
- Сглаживание
- Добавление случайных шумов
- Expanding mean

# ХЭШИРОВАНИЕ ПРИЗНАКОВ

- Если у категориального признака слишком много значений, скажем, миллион, то после применения one-hot кодировки мы получим миллион новых столбцов. С такой огромной матрицей тяжело работать.
- Хэширование развивает идею one-hot кодирования, но позволяет получать любое заранее заданное число новых числовых столбцов после кодировки.

# АЛГОРИТМ ХЭШИРОВАНИЯ

- 1) Для каждого значения признака вычисляем значение некоторой функции – хэш-функции (hash)
- 2) Задаем `hash_bucket_size` – итоговое количество различных значений категориального признака.
- 3) Берем остаток:  $\text{hash} \% \text{hash\_bucket\_size}$  – тем самым кодируем каждое значение признака числом от 0 до  $\text{hash\_bucket\_size}-1$ .
- 4) Далее к полученным числам применяем ONE.

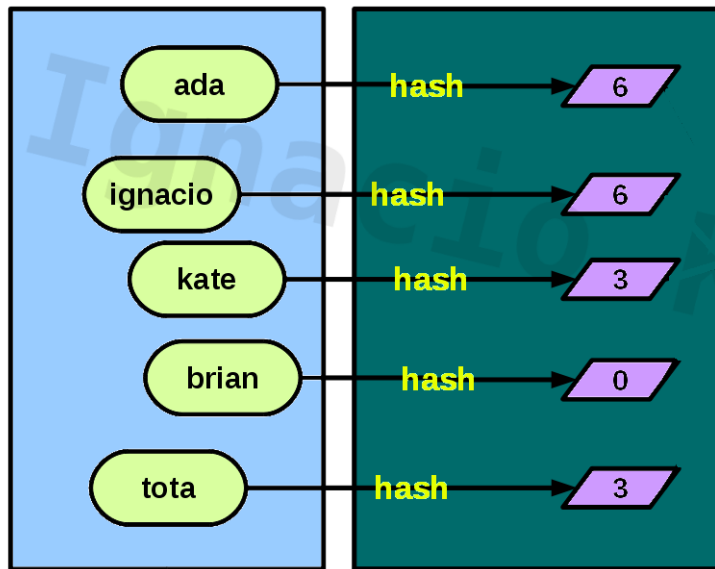


# ЧТО ДЕЛАЕТ ХЭШ-ФУНКЦИЯ

Идея: хэш-функция группирует значения категориального признака:

- часто встречающиеся значения признака формируют отдельные группы
- редко встречающиеся значения попадают в одну группу при группировке

# ХЭШИРОВАНИЕ ПРИЗНАКОВ: ПРИМЕР



elements

hash function

0	1	2	3	4	5	6
0	0	0	0	0	0	1
0	0	0	0	0	0	1
0	0	0	1	0	0	0
1	0	0	0	0	0	0
0	0	0	1	0	0	0

# ХЭШИРОВАНИЕ

- Хэширование – это способ кодирования категориальных данных, принимающих множество различных значений, показывающий хорошие результаты на практике.
- **Хэширование позволяет закодировать любое значение категориального признака (в том числе то, которого не было в тренировочной выборке).**

Статья про хэширование:

<https://arxiv.org/abs/1509.05472>

# ЧТО ПОЧИТАТЬ ПРО КОДИРОВАНИЕ КАТЕГОРИАЛЬНЫХ ПРИЗНАКОВ

- [Лекция Жени Соколова](#)
- [Блог Александра Дьяконова](#)
- [Кусочек статьи с Хабра про хеширование](#)