Happy Data Year

3'rd place solution

Хрыльченко Кирилл

2018

РОСБАНК Boosters.pro

Постановка задачи

Имеется выборка из 8765 объектов: 6261 тренировочных и 2504 тестовых.

- **Знаем:** адрес, координаты, группу банкомата.
- **Хотим узнать:** индекс популярности — некоторую величину, зависящую от частоты пользования банкоматом и (возможно) от размера денежных сумм операций.

Основные аспекты решения

- Корректировка данных
  - String matching
  - Одинаковые объекты с разным таргетом (надо что-то сделать)
- Сбор внешних данных
- Генерация признаков
- Обучение модели

Основные аспекты решения

- ~~Корректировка данных~~
  - ~~String matching~~
  - ~~Одинаковые объекты с разным таргетом (надо что-то сделать)~~
- Сбор внешних данных
- Генерация признаков
- Обучение модели

Внешние данные

- **Метро.** Три списка:
  - Москва: только координаты.
  - Санкт-Петербург: только координаты.
  - Россия: город, ветка, координаты.
- **Избирательные участки:** координаты, адрес, количество проголосовавших, тип здания, и т. п.
  - Адреса в едином формате — удобно извлекать информацию (город, регион)
- **Информация по городам.**
  - Таблица с github'а: город, координаты, регион, населенность.
  - Википедия: город, регион, населенность
  - Фед. служба гос. статистики: регион, населенность
  - Средние/минимальные зарплаты по регионам
  - И многое другое :)

Внешние данные

- **Парсинг:**
  - Регулярные выражения — боль и страдания
  - Красивый Супчик (aka Beautiful Soup)
- **Списки банкоматов:**
  - С сайтов Росбанка, Россельхозбанка, Газпромбанка, Райффайзенбанка
  - Таблицы с сайта Росбанка, Сбербанка
  - Сайт sravni.ru — около 40 тысяч банкоматов
  - ~~Альфабанк, ВТБ, Уралсиб Банк, АК Барс~~ (поленился)
- **OpenStreetMaps (OSM)** — все объекты в окрестности банкомата, имеющие хотя бы какие-нибудь тэги (свойства)

Feature Engineering

Имеем:

- $x_0 \in \mathcal{X}$ — банкомат
- M — множество объектов произвольной природы
- $\rho : \mathcal{X} \times M \to \mathbb{R}_+$ — расстояние

Что сработало:

- $\rho(x_0, M) := \min_{m \in M} \rho(x_0, m)$ — расстояние до ближайшего объекта
- Пусть $\rho_1 \leqslant \rho_2 \leqslant \cdots \leqslant \rho_m$ — расстояния до объектов из M. Тогда рассмотрим ряд $r_i := \rho_{i+1} - \rho_i$ — он стационарный (избавились от тренда) $\Rightarrow$ берем всевозможные статистики: mean, max, std, r_0.
- Статистики вычисляем не по всему ряду, а для некоторых $0 < k \leqslant m$, то есть только по $\rho_1 \leqslant \rho_2 \leqslant \cdots \leqslant \rho_k$.

Feature Engineering

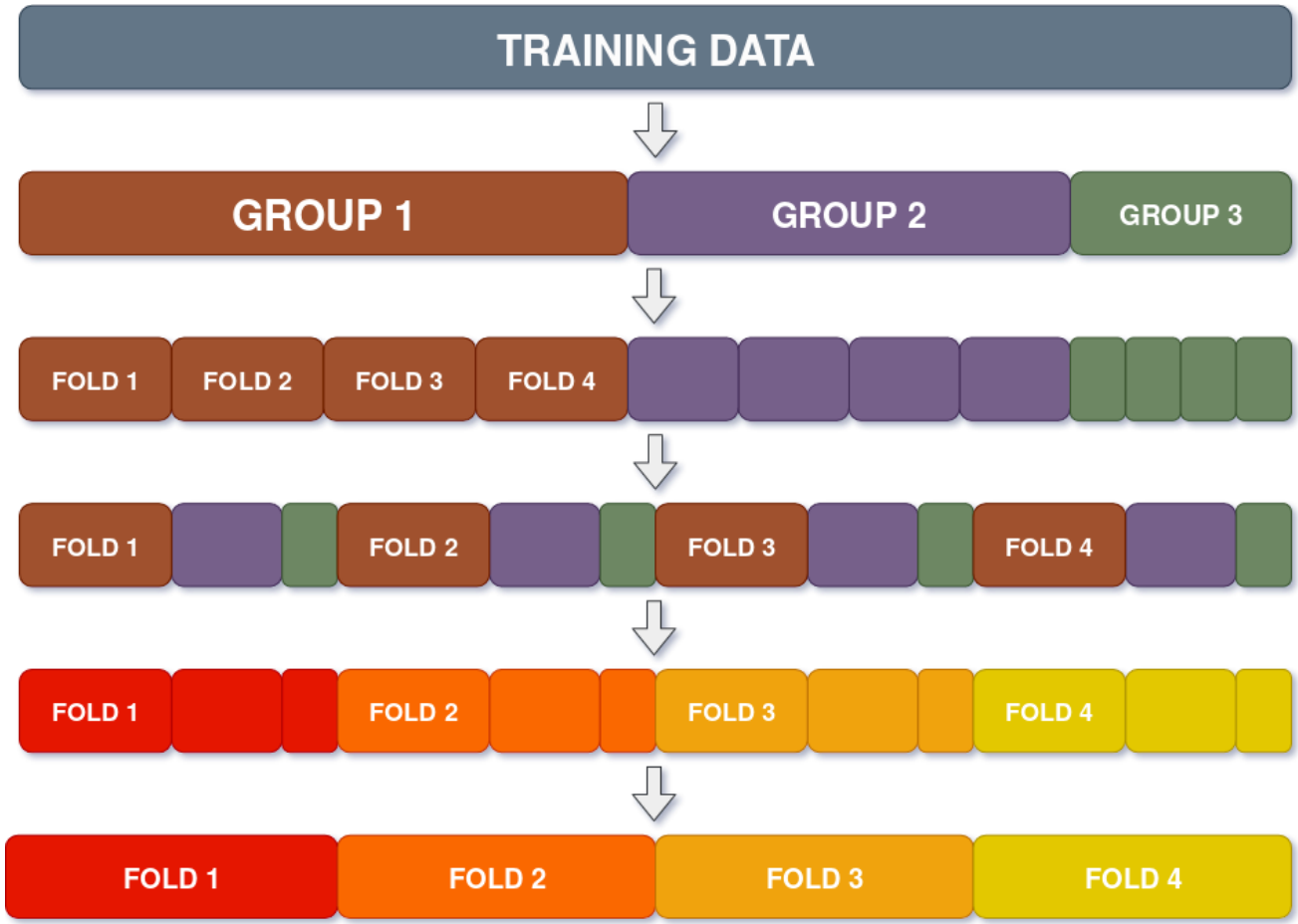
Пусть m_0 — ближайший к x_0 объект из M. Тогда все «характеристики» m_0 приписываем к x_0. Например:

- Банкоматы из sravni.ru: нашли ближайший банкомат $\Rightarrow$ используем время работы, услуги, валюты этого банкомата для нашего.
- Города: присваиваем население ближайшего города нашему банкомату.

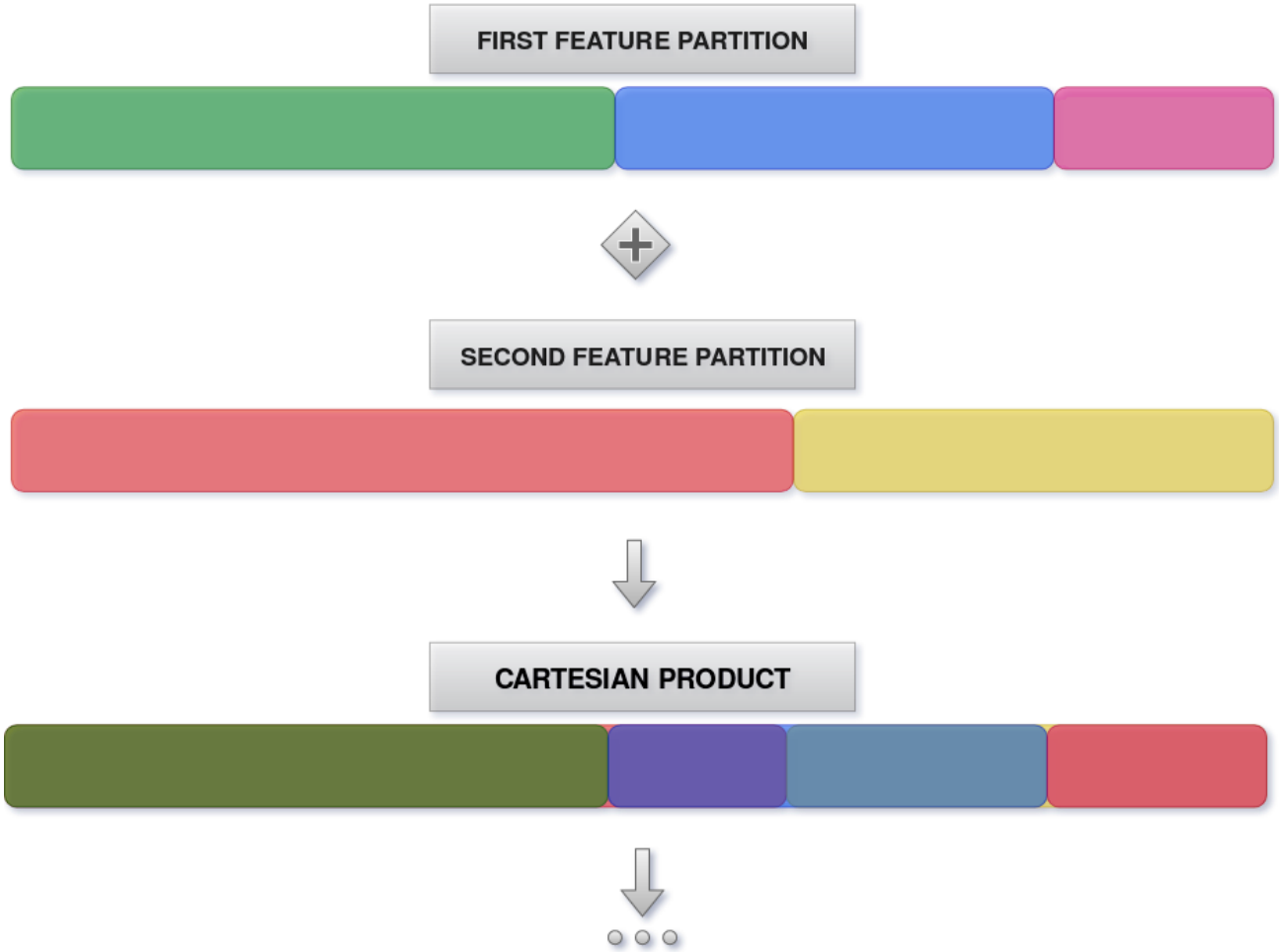
Кодирование категориальных признаков.

- Частота категории (количество объектов с такой же категорией)
- Количество объектов с такой же категорий и группой
- Доля объектов группы данного объекта среди всех объектов категории (отношение второй величины к первой)

Валидация. Стратификация по одному признаку



Валидация. Стратификация по двум признакам



Иерархическая валидация

При использовании валидации по нескольким признакам появляются слишком мелкие группы (количество объектов в группе меньше чем количество фолдов)

- **Простое решение:** собрать объекты из мелких групп и случайным образом равномерно раскидать по фолдам.
- **Иерархическая валидация:** упорядочить разделяющие признаки[1]. Выделить объекты из мелких групп. Откинуть последний, менее важный разделяющий признак, и рассмотреть деление на группы без него. Раскидать объекты из мелких групп по этому делению. Для оставшихся объектов снова откинуть последний признак, и т. п.

[1] Признаки, по которым проводится разбиение.

Модель

1. Используется библиотека для градиентного бустинга lightgbm, а конкретно LGBMRegressor
2. Иерархическая валидация на k фолдов по группе, региону и городу
3. Bagging — обучаются n_1 базовых моделей (LGBMRegressor) с разными seed'ами на одной и той же валидации (предсказания усредняются)
4. ValBagging — обучается n_2 Bagger'ов на разных seed'ах валидации.
5. Что делать с предсказаниями разных Bagger'ов?
   - Опять усреднение результатов
   - Сортировка по MSE, усредняем топ k моделей
   - Что-нибудь похитрее (to be continued)

Модель

Задача. Ваня из 7-го Б решил обучить модель со следующими параметрами:

- Валидация на 10 фолдов
- В каждом Bagger'е 6 базовых моделей
- В каждом ValBagger'е 30 Bagger'ов (Ваня знает, что на практике при ранней остановке так и будет)

Вопрос: сколько всего LGBMRegressor'ов придется обучить Ване?

Модель

Задача. Ваня из 7-го Б решил обучить модель со следующими параметрами:

- Валидация на 10 фолдов
- В каждом Bagger'е 6 базовых моделей
- В каждом ValBagger'е 30 Bagger'ов (Ваня знает, что на практике при ранней остановке так и будет)

Вопрос: сколько всего LGBMRegressor'ов придется обучить Ване?
Ответ: $10 \times 6 \times 30 = 1800$.

Модель

Задача. Ваня из 7-го Б решил обучить такую модель со следующими параметрами:

- Валидация на 10 фолдов
- В каждом Bagger'е 6 базовых моделей
- В каждом ValBagger'е 30 Bagger'ов (Ваня знает, что на практике при ранней остановке примерно столько и будет)

Вопрос: сколько всего LGBMRegressor'ов придется обучить Ване?
Ответ: $10 \times 6 \times 30 = 1800$. (не гвоздей, а LGBMRegressor'ов)
$\Rightarrow$ **Проблема:** Это очень много! При 400 признаках одна такая модель будет обучаться более часа.
$\Rightarrow$ **Решение:** Стохастический вариант модели!

Модель. Стохастический вариант

Сгруппируем признаки: метро, OSM, избирательные участки, sravni.ru, и т. п. (13 групп по 20 – 40 признаков)
На каждой внешней итерации (при обучении Bagger'а), т.е. при фиксированном seed'е валидации, будем семплировать $0 < m \leqslant 13$ групп (без возврата), и обучать Bagger только на них.
Результат:

- $m = 9 \Rightarrow$ примерно в 1.5 раза быстрее, чем «полный» вариант.
- Косвенная борьба с переобучением (модели проще, но разные)

Модель. Стохастический вариант

Сгруппируем признаки: метро, OSM, избирательные участки, sravni.ru, и т. п. (13 групп по 20 – 40 признаков)
На каждой внешней итерации (при обучении Bagger'а), т. е. при фиксированном seed'е валидации, будем семплировать $0 < m \leqslant 13$ групп (без возврата), и обучать Bagger только на них.
Результат:

- $m = 9 \Rightarrow$ примерно в 1.5 раза быстрее, чем «полный» вариант.
- Косвенная борьба с переобучением (модели проще, но разные)

$\Rightarrow$ **Проблема:** модели на внутреннем уровне (Bagger'ы) совсем разные по качеству, поэтому усреднять нерационально.
$\Rightarrow$ **Решение:** ~~что-то похитрее~~ ансамблирование.

Ensembling. Two models

Suppose we have:

- $y \in \mathbb{R}^m$ - target
- $p_1, p_2 \in \mathbb{R}^m$ — predictions of 2 different models

How to combine these predictions? **Optimization task:**

$$\frac{1}{2}\|\lambda p_1 + (1-\lambda)p_2 - y\|^2 \to \min_{\lambda \in [0,1]} \tag{1}$$

Solution based on **KKT**:

Let λ^* be $\frac{\langle p_1 - p_2, y - p_2 \rangle}{\|p_1 - p_2\|^2}$, then $\lambda = \begin{cases} 0, & \lambda^* \leqslant 0 \\ \lambda^*, & 0 < \lambda^* < 1 \\ 1, & \lambda^* \geqslant 1 \end{cases}$

Ensembling. Two models. Regularization

Main problem: overfitting (averaging may work better)
Solution: regularization

$$\frac{1}{2}\|\lambda p_1 + (1-\lambda)p_2 - y\|^2 + \frac{\alpha}{2}(1-\lambda)^2 \to \min_{\lambda \in [0,1]}$$

Solution:

Let λ^* be $\frac{\langle p_1 - p_2, y - p_2\rangle + \frac{\alpha}{2}}{\|p_1 - p_2\|^2 + \alpha}$, then $\lambda = \begin{cases} 0, & \lambda^* \leqslant 0 \\ \lambda^*, & 0 < \lambda^* < 1 \\ 1, & \lambda^* \geqslant 1 \end{cases}$

Sanity check: $\lambda \to_{\alpha \to \infty} 0.5$

Ensembling: n models

We have:

- $y \in \mathbb{R}^m$ - target
- $p_1, p_2, \ldots, p_n \in \mathbb{R}^m$ — predictions of n different models

Algorithm:

1. Solve (1) for $p_1, p_2 \Rightarrow$ get $\lambda_1 \Rightarrow$ apply λ_1: $p_{12} = \lambda_1 p_1 + (1 - \lambda_1) p_2$
2. Solve (1) for $p_{12}, p_3 \Rightarrow \ldots$
3. $\ldots$
4. Apply backwards procedure to get «absolute» values $\lambda_1^*, \ldots, \lambda_n^*$ from «relative» values $\lambda_1, \ldots, \lambda_n$.

Note: an additional trick for correct regularization.

Further modifications. Linear transformation

Let $p \in \mathbb{R}^m$ be predictions of our model (ensemble).
New task: find λ s.t. λp is better than just p:

$$\frac{1}{2}\|\lambda p - y\|^2 \to \min_{\lambda \in \mathbb{R}}$$

- **Solution:** $\lambda = \frac{\langle p, y \rangle}{\|p\|^2}$
- **Possible regularizer:** $\frac{\alpha}{2}(\lambda - 1)^2$
- Same task as previous one, but without constraints

Further modifications. Affine transformation

Let $p \in \mathbb{R}^m$ be predictions of our model (ensemble).
Task: find λ_1, λ_2 s.t. $\lambda_1 p + \lambda_2$ is better than p:

$$\frac{1}{2}\|\lambda_2 + \lambda_1 p - y\|^2 \to \min_{\lambda_1, \lambda_2 \in \mathbb{R}}$$

- **Solution:** $\lambda_1 = \frac{\|p\|^2\langle y, 1_m\rangle - \langle y, p\rangle\langle p, 1_m\rangle}{\|p\|^2 n - \langle p, 1_m\rangle^2}, \lambda_2 = \frac{\langle y, p\rangle - \lambda_1\langle p, 1_m\rangle}{\|p\|^2}$
- **Possible regularizer:** $\frac{\alpha_1}{2}(\lambda_1 - 1)^2 + \frac{\alpha_2}{2}\lambda_2^2$

Модель. Стохастический вариант. Улучшение

$\Rightarrow$ **Проблема:** модели на внутреннем уровне (Bagger'ы) совсем разные по качеству, поэтому усреднять нерационально.
$\Rightarrow$ **Решение:** ~~что-то похитрее~~ ансамблирование.
Будем считать каждый Bagger, который был обучен на своей подгруппе признаков, самостоятельной моделью. Коэффициенты в итоговой модели (в ValBagger'е) настраиваем по вышеописанной схеме, и используем раннюю остановку.
Новая проблема: сильное переобучение при большом количестве Bagger'ов.

Модель. Стохастический вариант

Новая проблема: сильное переобучение при большом количестве Bagger'ов.
Решение новой проблемы: подбор значения регуляризатора по кросс-валидации (еще одна кросс-валидация, внешняя, не путать с внутренней).
После «усреднения» всех моделей с помощью ансамблирования, применяется аффинное преобразование.

Модель. Стохастический вариант

Новая проблема: сильное переобучение при большом количестве Bagger'ов.
Решение новой проблемы: подбор значения регуляризатора по кросс-валидации (еще одна кросс-валидация, внешняя, не путать с внутренней).
После «усреднения» всех моделей с помощью ансамблирования, применяется аффинное преобразование.
Последняя фишка: усреднение и аффинное преобразование происходят «по группам»

Further modifications. Group transformations

Apply transformation to each group separately:

Group	λ	Old score	New score
5478	1.0445	0.00146470	0.00146117
1942	1.0988	0.00155497	0.00154666
8083	1.0142	0.00167942	0.00167474
496.5	1.0112	0.00204085	0.00204049
3185.5	1.0310	0.00234435	0.00234194
1022.0	1.0299	0.00120451	0.00119787
32.0	1.4359	0.00747002	0.00723829

Таблица: Groupwise linear transformation

Old score: 0.00168879, new score: 0.00168318

Спасибо за внимание!