О логическом подходе в задаче восстановления регрессии

Листопадов Иван Сергеевич

Московский государственный университет имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра математических методов прогнозирования

11 октября 2023 г.

Основные понятия

- M - множество объектов;
- $\{x_1, \ldots, x_n\}$ - множество признаков;
- $(x_1(S), \ldots, x_n(S))$ - признаковое описание объекта $S \in M$;
- Y - множество ответов;
- $y : M \to Y$ - целевая функция;
- $T = \{S_1, \ldots, S_m\}$ - обучающая выборка (прецеденты);
- $y_i = y(S_i)$ - значения целевой функции на прецеденте S_i;
- $A_T : M \to Y$ - алгоритм распознавания;
- H - набор различных признаков;
- $H(S)$ - признаковое подописание объекта S, определяемое набором признаков H.

Постановка задачи

Дано:

$T = \{S_1, \ldots, S_m\}, y_i = y(S_i), i = 1, ..m.$

Найти:

Некоторую вещественнозначную величину, т.е. $Y = \mathbb{R}$

Критерий:

Алгоритм $A_T : Q(A_T, T) \rightarrow min$, т.е. необходимо построить алгоритм который минимизирует функционал ошибки и наилучшим образом приближает зависимость между признаковыми описаниями объектов T и значениями целевой переменной Y.

Линейная регрессия. Сведение к оптимизационной задаче

1. Для описания зависимости целевой переменной Y от признаковов $x_1, \ldots, x_n$ используется линейная модель:
$$Y = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n,$$

2. Метод обучения - метод наименьших квадратов (МНК):
$$\sum_{j=1}^{m} \left[y_j - \beta_0 - \sum_{i=1}^{n} x_i \beta_i \right]^2$$

3. Проверка по тестовой выборке X^k:
$$Q(X^k) = \frac{1}{k} \sum_{i=1}^{k} \left[\widetilde{y_j} - \beta_0 - \sum_{i=1}^{n} \widetilde{x_i} \beta_i \right]^2$$

Метрические методы регрессии (kNN-регрессия)

Обучение:

$d(.,.)$– функция расстояния, например, евклидово расстояние, тогда

$$d(S_k, S_p) = \sqrt{\sum_{j=1}^{n}(x_j(S_k) - x_j(S_p)^2}.$$

Отладка (подбор гиперпараметра k):

$$\text{Выбор оптимального } k: k^* = \arg\min_k \frac{1}{m}\sum_{i=1}^{m}\left(y_i - \frac{1}{k}\sum_{i=1}^{k} y_{\text{сосед}}\right)^2.$$

Прогнозирование:

Найти k ближайших соседей для $S_0: d(S_0, S_0^1) \leqslant d(S_0, S_0^2) \leqslant \ldots \leqslant d(S_0, S_0^k)$;

$$\text{Прогноз: } \hat{y} = \frac{1}{k}\sum_{i=1}^{k} y_{\text{сосед}}$$

Предложенный алгоритм: A1-Rg

Разметка обучающего набора данных с помощью кластеризации

$$\mathcal{Y} = \text{DBSCAN}(X, \epsilon) \tag{1}$$

Классификация по прецедентам

- Поиск тупиковых представительных элементарных классификаторов:

$$h_i = \arg\min_h \sum_{y_j \neq y_i} w_j \cdot I(h(x_j) \neq y_j) \tag{2}$$

Прогнозирование

- Восстановление значения целевой переменной:

$$y_{new} = \frac{\sum_{x_j \in C_{new}} y_j \cdot w_j}{\sum_{x_j \in C_{new}} w_j}$$

Описание экспериментальной части

- **Цель экспериментов** - сравнить алгоритм A1-Rg с классическими методами восстановления регрессии:
  - Линейная регрессия (LR)
  - kNN-регрессия (kNR)
  - Градиентный бустинг (GB)
  - Регрессия с помощью метода опорных векторов (SVR)
  - Случайный лес (RF)
- **Предобработка данных**:
  - Кодирование вещественных признаков в категориальные
  - Удаление колонок с большим количеством уникальных значений
  - Удаление объектов с пропущенными значениями
  - Кодирование категориальных признаков в числовые значения
- **Оценка моделей**: Кросс-валидация с 5 фолдами, метрики оценки: RMSE и коэффициент детерминации R2, 10 независимых запусков для более точной оценки.

Результаты экспериментов

Таблица 2: Качество (R2 Score)

Данные	LR	GB	kNR	RF	SVR	A1-Rg
Automobile (201 × 12)	0.73	**0.90**	0.71	0.88	0.35	0.36
Computer Hardware (209 × 9)	0.69	**0.86**	0.38	0.85	0.62	0.19
Servo Data (167 × 6)	0.45	**0.91**	0.66	0.90	0.81	0.27
Yacht Hydrodyn. (308 × 7)	0.63	0.86	0.76	**0.90**	0.83	0.66

Ансамбль моделей. Бустинг

Инициализация:

- Обучающая выборка - T, количество базовых моделей - N, $F_0(x) = \text{const}$ - начальное приближение ответа.

Идея - обучение по остаткам:

1. На n-ом шаге вычисляется остаток на предыдущей итерации:

$$r_{n-1}(x_i) = y_i - F_{n-1}(x_i)$$

2. Строится базовая модель $b_n(x)$, которая описывает остаток:

$$b_n(x) = \arg\min_b \sum_{i=1}^{m} L(r_{n-1}(x_i), b(x_i))$$

3. К базовой модели добавляется с весом α_n, формируя композицию:

$$F_n(x) = F_{n-1}(x) + \alpha_n b_n(x)$$

4. Обновляется ответ:

$$y_i = F_n(x_i)$$

Предложенный алгоритм: бустинг над эл.кл.

1. На каждой итерации (t) строим слабый алгоритм h_t вычисляя остатки при решении задачи регрессии:

$$h_t = \arg\min_h \sum_{S_{l_j} \in D} [H(S_{i_t}) = H(S_{l_j})] \cdot (y_{i_t} - y_{l_j})^2$$

2. Вычисляем веса α_t для слабого алгоритма h_t:

$$\alpha_t = \arg\min_\alpha \sum_{i \neq i_t} \left| y_{i_t} - y_i \right|$$

3. Обновляем остатки, учитывая вклад каждого слабого алгоритма:

$$r_i = y_i - \sum_{t=1}^{T} \alpha_t \cdot h_t(x_i)$$

4. Выбираем объекты с наибольшими остатками r_i для построения следующего слабого алгоритма:

$$S_i = \{x_i\}_{i=1}^{r}, \text{ где } r \text{ - параметр алгоритма}$$

Бустинг над эл.кл. (продолжение)

5. Построение булевой матрицы сравнения L_t для итерации t:

$$L_t = \left\| a_{lj} \right\|, i \in \{1, \ldots, r\}, j \in \{1, \ldots, n\},$$

в которой

$$a_{lj} = \left[x_j \left(S_{k_l} \right) \neq x_j \left(S_{i_t} \right) \right],$$

то есть в каждой строке l индикаторы того, по каким признаком можно различить объекты S_{k_l} и S_{i_t}

6. Нахождение неприводимого покрытия H_t матрицы L_t с использованием квадратичного функционала.

7. Слабый алгоритм h_t использует набор признаков H_t и объект x_i для построения предсказания:

$$h_t(x_i) = [H(S_{i_t}) = H(S_{l_j})] \cdot f(y_{i_t})$$

Слайд о будущей работе

Исследовать методы решения задачи восстановления регрессии с применением дискретных процедур.

Цель работы —

Построить оптимальный алгоритм восстановления регрессии с точки зрения выбранной метрики качества на базе дискретных процедур распознавания и экспериментально сравнить различные подходы.

Необходимо реализовать

Алгоритм бустинга над элементарными классификаторами с использованием описанной схемы работы.