ЭКСПЕРТНЫЕ СИСТЕМЫ И РАСПОЗНАВАНИЕ ОБРАЗОВ
Вычислительные системы
1988 год Выпуск 126

УДК 519.237

ЛОКАЛЬНЫЕ МЕТОДЫ ЗАПОЛНЕНИЯ ПРОБЕЛОВ В ЭМПИРИЧЕСКИХ ТАБЛИЦАХ

Н.Г.Загоруйко, Г.В.Ульянов

В настоящей работе рассматриваются локально-параметрические и непараметрические методы заполнения пробелов в таблицах данных типа "объект-свойство", предназначенные для подготовки таблиц к применению классических процедур анализа данных, для выявления аномальных наблюдений или ошибок и непосредственно для оценивания отсутствующих значений в таблицах.

§ 1. Задача прогнозирования пробелов

Определим понятия "зависимость" и "прогноз" в рамках теории статистического прогнозирования [12,с.234; 14]. Для этого выделим в таблице данных X $(m \times n)$ какой-либо пробел ("основной", остальные пробелы - "*сопутствующие*") и введем для него понятие "оптимального прогноза". Пусть этот пробел лежит в столбце y (который назовем *откликом*) и строке $z_0 = (y_0, \underline{x}_0)$. Здесь y_0 - элемент строки, лежащий в столбце y (пробел), а $\underline{x}_0$ - вектор прочих элементов строки. Остальные столбцы $x_1 \ldots x_n$ назовем *предикторами*. Нам нужен алгоритм A: $A[X, y_0] = \hat{g}$, который каждой таблице X и каждому пробелу y_0 в ней ставил бы в соответствие некоторую *прогнозирующую функцию* $\hat{g}(\underline{x})$, определенную в некоторой ок-

рестности $\underline{x} \in U(\underline{x}_0)$ точки $\underline{x}_0$ так, чтобы для $\forall \underline{x} \in U(\underline{x}_0)$ эта функция давала бы прогноз $\hat{y} = \hat{g}(\underline{x})$, в том числе и для пробела y_0: $\hat{y}_0 = \hat{g}(\underline{x}_0)$. Столбцам таблицы X поставим в соответствие случайные величины $y, x_1 \ldots x_n$. Будем считать, что строки таблицы $z_i = (y_i, \underline{x}_i)$, $i = 1, \ldots, m$, независимо и одинаково распределены с некоторой неизвестной плотностью $p(y, \underline{x})$. Пусть далее для прогноза используется некоторая функция $\hat{g}(\underline{x})$ из класса $\mathcal{L}$. Определим *функцию потерь* ρ от неверного предсказания значений y функцией $\hat{g}(\underline{x})$. В настоящей статье $\rho(y, \hat{g}) = (y - \hat{g}(\underline{x}))^2$. Введем средние потери от предсказания:

$$W(\hat{g}) = E\rho(y, \hat{g}(\underline{x})) = \int (y - \hat{g}(\underline{x}))^2 p(y, \underline{x}) dy d\underline{x},$$

где E - символ матожидания.

Будем считать оптимальной прогнозирующей функцией ту, которая минимизирует этот критерий. Известно, что этот минимум достигается на функции $g(\underline{x}) = E(y|\underline{x})$, т.е. на теоретической регрессии (если $g(\underline{x}) \in \mathcal{L}$). В дальнейшем, говоря о (теоретической) *зависимости* будем иметь в виду функцию $g(\underline{x})$. Известно также, что значение $W(\hat{g})$ для некоторой $\hat{g}(\underline{x})$ тем меньше, чем ближе $\hat{g}(\underline{x})$ к $g(\underline{x})$ в метрике L_2. Поэтому в качестве прогнозирующей функции $\hat{g}(\underline{x})$ будем выбирать ту или иную оценку зависимости $g(\underline{x})$. *Оптимальным прогнозом* будем считать значение $\hat{y}_0 = g(\underline{x}_0)$.

Относительно механизма возникновения пробелов в таблице будем вслед за [32] предполагать, что данные отсутствуют "совершенно случайно", т.е. индикатор отсутствия элемента таблицы как случайная величина не зависит от значения как этого элемента, так и любых других элементов таблицы. Кроме того, индикаторы не зависят друг от друга.

§ 2. Обзор последних работ

Рассмотрим вкратце историю развития исследований, связанных с обработкой пробелов. В докомпьютерное время (до 1960 г.), начиная со статьи Уилкса [40], исследования в этой области носили в основном теоретический характер и касались большей частью оценок максимального правдоподобия (МП-оценки) по некомплектным выборкам.

На практике же использовались примитивные способы борьбы с пробелами - вычеркивание некомплектных строк или столбцов, замена пробелов средними по столбцу, использование в вычислениях только комплектных пар и т.п. Полный обзор этих и многих других методов до 1966 г. можно найти в [15], некоторые из них приведены в [2,с.192].

С распространением ЭВМ были предложены более сложные машинные алгоритмы, основанные на методе наименьших квадратов: регрессионный метод [18,39], метод главных компонент [24], пошаговая регрессия [22], метод многомерной линейной экстраполяции [13], метод прогностических переменных [7]. Учитывая тот факт, что оценки первых двух моментов полностью определяют оценки регрессии, многие авторы сосредоточились на проблеме оценивания ковариационной матрицы по данным с отсутствующими значениями [21,23,28]. Одновременно выяснилась и некоторая ограниченность методов, основанных на методе наименьших квадратов. Так, Уилкинсон [9,с.167] указывал, что если пробелы имеются только в отклике, то при совместном оценивании пробелов и коэффициентов регрессии метод наименьших квадратов требует вычеркивать все строки с пробелами. Это приводит к неполному использованию информации, содержащейся в данных.

Со второй половины 70-х годов особых успехов добилось направление, связанное с МП-оценками, особенно в рамках нормальных распределений. Появились практические алгоритмы, вычисляющие МП-оценки пробелов, например [25,17,37]. В работе [19]

предложена мощная вычислительная процедура - EM-алгоритм для решения общей задачи оценивания параметров в условиях некомп - лектной выборки. К настоящему времени эти методы интенсивно развиваются, созданы эффективные робастные варианты EM-алгоритма [31]. Возобладала тенденция поиска для всех классических статистических методов аналогов, способных работать с некомплектными данными, не заполняя пробелов [27,29,34]. Более пол - ный обзор теории и практики содержится в монографиях [20,30].

§ 3. Локально-параметрические и непараметрические методы заполнения пробелов

Методы, упоминавшиеся в обзоре, действуют, как правило, глобально, т.е. в них предполагается, что зависимость заданного (например, линейного) типа реализована на всех объектах,поэтому и в оценивании зависимостей участвуют все строки.Локально-параметрические алгоритмы, оценивающие зависимость по некомплектной выборке в некоторой окрестности предсказываемого объекта, а также непараметрические алгоритмы оценивания регрессии по данным с пробелами развиты пока слабо. Более того,даже непараметрические оценки плотности в условиях некомплектной выборки почти не рассматривались [38].

По-видимому, первым локально-параметрическим алгоритмом заполнения пробелов можно считать ZET [8] (последняя версия [6,с.85]). В данном алгоритме откликом и предикторами могут быть как столбцы, так и строки. В справочнике [1,с.414] он отнесен к непараметрическим алгоритмам. Однако прогноз локальным средним по отклику, как это определено в книге, применяется лишь в вырожденных ситуациях, в общем же случае прогнозирующей функцией является взвешенное среднее простых линейных регрессий отклика на часть предикторов. Это больше соответствует параметрической модели, так как известно, что метод наименьших квадратов при отличии распределения ошибок от

нормального уже не приводит к эффективным оценкам зависимости.

В настоящей работе продолжается линия развития метода ZET. Разработано семейство алгоритмов с общим именем ZL, два из которых ZL-ПШ ("пошаговый") и ZL-СТ ("ступенчатый") являются локально-параметрическими, а третий - ZL-НЕП ("непараметрический", с вариантами ZL-НСПА, ZL-НПШ, ZL-НСТ) дает непараметрическую оценку зависимости. Отличия новых алгоритмов от ZET следующие: 1) ZET рассчитан на оценивание парных зависимостей, ZL - на оценивание множественных; 2) в ZL принята квадратическая функция потерь ρ, а в ZET $\rho(y,\hat{g}(\underline{x})) = |y-\hat{g}(x)|/|y|$. При минимизации этой функции возникают трудности, связанные с негладкостью ρ, а также, когда y близок к нулю. Далее, в ZL: 3) другая нормировка столбцов; 4) другой способ отбора информативных столбцов; 5) другой способ обработки сопутствующих пробелов; 6) другая модель регрессии. Эти отличия будут рас - шифрованы в ходе изложения алгоритма.

§ 4. Алгоритмы семейства ZL. Общая часть

Рассмотрим работу алгоритмов из ZL-семейства на какой-либо таблице $X'(m \times n)$, $m \geq 3, n \geq 1$. В ZL каждый пробел или редактируемый элемент обрабатывается *независимо*, т.е. информация о заполнении предыдущих пробелов не используется для заполнения данного пробела; кроме того, характеристики столбцов при прогнозе каждого пробела вычисляются заново. Это связано с локальностью этих характеристик. В глобальных же алгоритмах рациональнее разбивать строки на группы с одинаковыми "образами" пробелов и находить единую прогнозирующую функцию для каждого столбца и каждой группы. В алгоритме ZET есть другие возможности. Так или иначе, достаточно описать процесс заполнения алгоритмом ZL какого-нибудь одного пробела.

В ZL применяется следующий способ *обработки сопутствующих пробелов:* первичное заполнение пробелов средними по столбцу, оценка регрессий по укомплектованным данным и затем вторичное заполнение посредством регрессии. Этот способ ("метод регрессии 1-го порядка") предложен в [15], а затем развит в [26]. В первой работе показано, что в условиях слабой средней корреляции в таблице и/или при значительном числе пробелов данный подход эффективнее классического вычеркивания строк. В ZET же все оценки находятся только по комплектным парам элементов.

Пусть прогноз требуется для элемента $x'_{i_0 j_0}$. Назовем строку i_0 *предсказываемой*, а столбец j_0 - *предсказываемым* (откликом). Вначале изложим общую часть всех трех алгоритмов из семейства ZL, а затем их специфические части. Так же, как и в ZET, пользователь должен вначале задать размеры *предсказывающей подматрицы* B (см. ниже) $(m_0+1)\times(n_0+1), m_0 \geq 2, n_0 \geq 1$, на которой и будет производиться оценивание зависимости. Пробелы в таблице заменяем уникальным числом PROB, на несколько порядков большим элементов таблицы.

Пусть M'_k - множество индексов отличных от пробела элементов k-го столбца: $M'_k=\{1 \leq i \leq n \mid i \neq i_0, x_{ik} \neq \mathrm{PROB}\}$, $|M'_k| = m'_k$, $k = 1,\ldots,n$. Все характеристики таблицы X' будут иметь штрих в обозначениях. Предсказываемая строка не включается в M'_k. Столбцы с $m'_k = 0$ помечаем как *глобально-пустые* (при этом элемент i_0 такого столбца может не равняться PROB). Если $x_{i_0 k} \neq \mathrm{PROB}$, то k-й столбец будем называть *инцидентным* (предсказываемой строке). Глобально-пустой столбец может быть инцидентным. Далее через Всi, $i = 1,\ldots,6$, будем обозначать вырожденные ситуации, прекращающие работу алгоритма.

ВС1. Если $m'_{j_0} = 0$ (отклик глобально-пуст) - отказываемся от прогноза данного пробела.

Для глобально-непустых столбцов $(m'_k > 0)$ вычисляем глобальные средние $\bar{x}'_k$ и глобальные среднеквадратические отклонения s'_k:

$$\bar{x}'_k = \frac{1}{m'_k} \sum_i x'_{ik}; \quad s'_k = \left\{ \frac{1}{m-2} \sum_i (x'_{ik} - \bar{x}'_k)^2 \right\}^{1/2}; \quad i \in M'_k.$$

Если $m'_k = 0$, то $\bar{x}'_k = s'_k = PROB$. Столбцы с $s'_k < \varepsilon$ помечаем как *глобально-вырожденные*. Здесь ε - некоторая малая константа, задаваемая пользователем (например, 10^{-35}). Обозначим $\bar{y}' = \bar{x}'_{j_0}$; $s'_y = s'_{j_0}$; K' - множество индексов глобально-невырожденных глобально-непустых инцидентных столбцов, $N' = |K'|$.

ВС2. Если $s'_y = 0$ (отклик глобально-вырожден), то прогноз пробела $\hat{y}_0 = \bar{y}'$, а ожидаемая ошибка $G = 0\%$.

ВС3. Если $N' = 0$, то прогноз $\hat{y}_0 = \bar{y}'$, а $G = 100\%$. Эта ситуация имеет место, в частности, когда предсказываемая строка пуста.

Далее нормируем глобально-невырожденные столбцы, т.е. переходим от таблицы X' к таблице $X = \{x_{jk}\}$ $(m \times n)$ с элементами без штриха: $x_{ik} = (x'_{ik} - \bar{x}'_k)/s'_k$, если $s'_k \geq \varepsilon$; $i = 1,\ldots,m$; $k = 1,\ldots,n$. Если $x'_{ki} = PROB$, то и $x_{ik} = PROB$. Вырожденные и пустые столбцы не нормируются и вообще не используются. Характеристики таблицы X будут отличаться верхним индексом "0" вместо штриха. После нормировки $s^0_k = 1$, $\bar{x}^0_k = 0$ для глобально-невырожденных столбцов $(s_k > \varepsilon)$.

Традиционная нормировка по дисперсиям обладает неприятным свойством портить столбцы при наличии в них аномальных значений, превращая все элементы в 0 и 1. Можно предложить новую

медианную нормировку, свободную от этого недостатка: надо вычислять вместо $\bar{x}'_k$ медианы: $\mathrm{med}'_k = \underset{i}{\mathrm{med}}\, x_{ik}$, $i \in M'_k$, а вместо s'_k - абсолютные медианные отклонения: $\mathrm{AMO}'_k = \underset{i}{\mathrm{med}}\,|x'_{ik} - \mathrm{med}'_k|$, $i \in M'_k$; формула же для нормировки имеет вид:

$$x_{ik} = (x'_{ik} - \mathrm{med}'_k)/\,\mathrm{AMO}'_k, \quad \text{если} \quad \mathrm{AMO}'_k \geq \varepsilon .$$

При этом в вырожденных ситуациях фигурирует med'_k вместо $\bar{x}'_k$ и AMO'_k вместо s'_k, а прогноз приобретает свойство устойчивости с пороговой точкой, равной $(m-m_0)/(2m)$, если загрязнение не касается предсказываемой строки. Правда, при этом медианная нормировка требует больше времени.

Далее, назовем i-ю строку *инцидентной* (отклику), если $x_{ij_0} \neq \mathrm{PROB}$. Предсказываемая строка и отклик являются взаимно неинцидентными. Выберем заданное число m_0 инцидентных строк, ближайших в смысле глобального евклидова расстояния $r^0_{ii_0}$ к предсказываемой строке i_0:

$$r^0_{ii_0} = \Big\{ \sum_{k \in K'} (\tilde{x}_{ik} - \tilde{x}_{i_0 k})^2 \Big\}^{1/2};$$

$$\tilde{x}_{ik} = \begin{cases} x_{ik}, & x_{ik} \neq \mathrm{PROB}; \\ \bar{x}^0_k, & x_{ik} = \mathrm{PROB}. \end{cases}$$

Если m_0 инцидентных строк нельзя найти, добавляем неинцидентные, ближайшие к строке i_0. Так как столбец j_0 глобально непуст и неинцидентен, должна существовать хотя бы одна инцидентная строка $i \neq i_0$. Пусть отобраны строки с индексами $i \in I$, $I = \{i_1, \ldots, i_{m_0}\}$, $|I| = m_0$. Вместе со строкой i_0 они образуют *матрицу выбранных строк* C $(m_0+1) \times n$.

Характеристики столбцов матрицы C (*локальные* характеристики таблицы) будут отличаться отсутствием всякого верхнего индекса.

Будем записывать столбцы матрицы C - $x_1,\ldots,x_n$, отклик - $y = x_{j_0}$, предсказываемую строку $c_0 = (y_0, \underline{x}_0)$, остальные строки $c_1 = (y_1, \underline{x}_1), \ldots, c_{m_0} = (y_{m_0}, \underline{x}_{m_0})$. Столбец x_k матрицы C помечаем как *локально-пустой*, если $m_k = 0$, где $m_k = |M_k|$, $M_k = \{i \in I \mid x_{ik} \neq PROB\}$. Находим локальные средние $\bar{x}_k$ для локально-непустых столбцов:

$$\bar{x}_k = \frac{1}{m_k} \sum_i x_{ik}, \quad i \in M_k; \quad m_k \neq 0, k = 1,\ldots,n.$$

Заполняем пробелы в локально-непустых столбцах матрицы C локальными средними $\bar{x}_k$ (*первичное заполнение*). Для укомплектованных таким образом столбцов вычисляем локальные среднеквадратические отклонения s_k:

$$s_k = \left\{\frac{1}{m_0 - 1} \sum_{i \in I} (x_{ik} - \bar{x}_k)^2\right\}^{1/2}; \quad m_k \neq 0, \; k = 1,\ldots,n.$$

Если $m_k = 0$, то $\bar{x}_k = s_k = PROB$. Столбцы с $s_k < \varepsilon_0$ будем называть *локально-вырожденными*. Здесь ε_0 - константа (например, 10^{-5}).

Введем обозначения: K - множество номеров локально-непустых локально-невырожденных инцидентных столбцов матрицы C; $s_y = s_{j_0}$: $\bar{y} = \bar{x}_{j_0}$. Если имеет место хотя бы одна из следующих ситуаций:

<u>ВС4</u>. $s_y < \varepsilon_0$ (отклик локально-вырожден);

<u>ВС5</u>. $N = 0$, где $N = |K|$;

<u>ВС6</u>. $m_0 < 3$,

то прогноз $\hat{y}_0 = \bar{y}$, а ожидаемая ошибка $G = (s_y / s_y^0) \cdot 100 = (s_y \cdot 100)\%$.

Далее следует описание собственно процедур прогнозирования, которые базируются на трех различных моделях регрессии и, кроме того, отличаются способами отбора подмножества предикторов J для предсказывающей подматрицы $B = \{x_{ij}\}$, $i \in I \cup \{i_0\}$, $j \in J \cup \{j_0\}$, а также методами оценивания локальной зависимости отклика y от предикторов $x_{j_1}, \ldots, x_{j_{n_0}}$, $J = \{j_1, \ldots, j_{n_0}\}$. Сначала опишем локально-параметрические алгоритмы ZL-СТ и ZL-ПШ, а затем непараметрический алгоритм ZL-НЕП. Переход к этим процедурам происходит только в том случае, если никакая вырожденная ситуация не имела место.

§ 5. Алгоритм ZL-СТ

В данном алгоритме для прогноза используется ступенчатый метод оценки множественной линейной регрессии [5, с. 53]. Предполагается, что $g(\underline{x}) = E(y|\underline{x}) = \alpha^T \underline{x} + \alpha_0$. Этот метод дает решение, близкое к точным оценкам по методу наименьших квадратов, когда предикаты слабо коррелированы; в противном случае оценки, как правило, менее эффективны, чем оценки по методу наименьших квадратов. Пользователь должен задать значения параметров n_0, g_0, ω (см. ниже).

Оптимальный набор предикторов J строится итерационно. К выбору допускаются лишь столбцы из K. Будем обозначать через J_t множество столбцов, отобранных на шаге t, через $e(t)$ — вектор остатков на этом шаге. Полагаем вначале $J_0 = \emptyset$, $e(0) = y$.

Шаг 1. Находим столбец $x_{j_1}: |\rho(y, x_{j_1})| = \max_j |\rho(e(0), x_j)|$, $j \in K \setminus J_0$, ближайший к отклику в смысле $|\rho(\cdot)|$ — модуля локального коэффициента корреляции (см. ниже). Обозначим $\rho_1 = \rho(y, x_{j_1})$. Если $|\rho_1| \le \omega$, где ω — некоторый порог (например, 0,5-0,6), то прогноз пробела определяется $\hat{y}_0 = \bar{y}$,

$G = (s_y \cdot 100)\%$. Если же $|\rho_1| > \omega$, вычисляем вектор прогнозов $\hat{y}(1)$ для элементов отклика y по столбцу x_{j_1} согласно уравнению регрессии, а также вектор остатков $e(1)$ (коэффициенты регрессии α_1 и β_1 определяются ниже):

$$J_1 = J_0 \cup \{j_1\}; \quad d(1) = \alpha_1 x_{j_1} + \beta_1;$$

$$\hat{y}(1) = d(1); \qquad e(1) = y - d(1).$$

Ожидаемая ошибка прогноза на первом шаге будет:

$$G_1 = 100\,\|e(1)\|/(m_0-2)^{1/2}\,\% \;, \quad \|e(1)\| = \sum_{i\in I} e_i^2(1).$$

Если $G_1 < g_0$, где g_0 - заданный порог точности (например, 1-2%), то прекращаем дальнейший отбор столбцов, так как достигнут требуемый уровень точности. В противном случае переходим к следующему шагу.

Шаг $t \le n_0$. Пусть $J_{t-1} = \{j_1, \ldots, j_{t-1}\}$. Ищем столбец x_{j_t}:

$$|\rho(x_{j_t}, e(t-1))| = \max_j |\rho(e(t-1), x_j)|; \; j \in K \setminus J_{t-1},$$

т.е. наиболее коррелированный с вектором остатков $e(t-1)$. Такой столбец можно не найти по следующим причинам:

а) $K \setminus J_{t-1} = \emptyset$, б) $s_u < \varepsilon_0$,

где $u = e(t-1)$ - вектор остатков локально-вырожден;

в) $|\rho_t| = |\rho(e(t-1), x_{j_t}| < \omega$ -

вектор остатков слабо коррелирован с остальными столбцами. Если одна из этих ситуаций имеет место, то отбор столбцов прекращается, а среди сделанных прогнозов определяется оптимальный (см. ниже). В противном случае вычисляется регрессия вектора остатков $e(t-1)$ на очередной столбец j_t и определяется новый вектор прогнозов $\hat{y}(t)$ (здесь $k = j_t$):

$$J_t = J_{t-1} \cup \{ x_k \}; \quad d(t) = \alpha_t x_k + \beta_t;$$

$$\hat{y}(t) = y(t-1) + d(t); \quad e(t) = e(t-1) - d(t).$$

Ожидаемая ошибка на шаге t :

$$G_t = 100\|e(t)\|/(m_0 - t - 1)^{1/2} \%; \quad \|e(t)\| = \sum_{i \in I} e_i^2(t).$$

Если $G_t < g_0$, то прекращаем дальнейший отбор столбцов.

Итерации производятся для $t = 1, \ldots, n_0$, но могут закончиться и раньше, при некотором $t^* = t^*(g_0)$. Оптимальным прогнозом считается прогноз, сделанный на шаге $t' = \arg\min_t G_t$, $t = 1, \ldots, t^*$. Оптимальный набор столбцов $J = J_{t'}$; окончательный прогноз $\hat{y}_0 = \hat{y}_0(t')$; ожидаемая ошибка $G = G_{t'}$. Выше используются следующие формулы ($k = j_t$, $u = e(t-1)$):

а) регрессия вектора остатков u на новый столбец x_k:

$$d_i(t) = \bar{u} + \rho_t \frac{s_u}{s_{x_k}} (x_{ik} - \bar{x}_k) = \alpha_t x_{ik} + \beta_t;$$

б) локальный коэффициент корреляции u с x_k:

$$\rho(u, x_k) = \Big(\sum_{i \in I} u_i x_{ik} - m_0 \bar{u} \bar{x}_k \Big) / ((m_0 - 1) s_u s_{x_k});$$

здесь

$$\bar{u} = \Big(\sum_{i \in I} u_i \Big)/m_0; \quad s_u^2 = \Big(\sum_{i \in I} u_i^2 - m_0 \bar{u}^2 \Big)/(m_0 - 1).$$

§ 6. Алгоритм ZL-ПШ

В данном алгоритме реализован пошаговый метод оценки линейной регрессии. Этот метод дает точное решение по методу наименьших квадратов, правда, на приблизительно оптимальном наборе предикторов. Изложение метода с некоторыми незначительными модификациями следует работе [4].

Для заполнения пробелов алгоритмы, аналогичные данному, используются особенно широко. Если игнорировать локальность оценок и проблему сопутствующих пробелов, ZL-ПШ можно считать первой итерацией частного случая EM-алгоритма (случая нормально распределенных данных) [19]. Близок он также к методу многомерной линейной экстраполяции [13, с.49]. В последней, как и в ZL, провозглашается принцип локальности, отличается же он от ZL следующим. В методе многомерной линейной экстраполяции другой - "оптимизационный" способ обхода сопутствующих пробелов; другая процедура отбора строк: в подматрице должно быть $m_0 \leq n_0$ строк; отсутствует процедура отбора предикторов.

Перейдем к изложению алгоритма. Пользователь должен задать значения параметров F_{in}, F_{out}, TOL, n_0 (см. ниже).

Вначале вычисляется локальная матрица перекрестных произведений A:

$$a_{jl} = \sum_{i \in I} (x_{ij} - \bar{x}_j)(x_{il} - \bar{x}_l);\ j,l \in K \cup \{j_0\}.$$

Далее организуется итерационный процесс построения множества J, на каждом шаге которого оцениваются коэффициенты регрессии, делается прогноз и вычисляется ошибка прогноза G. Для перехода от шага к шагу используется так называемый оператор выметания Эфроимсона. На каждой итерации делается либо шаг вперед (в J включается очередной столбец), либо шаг назад (исключается). На 1-й итерации - только шаг вперед. Множество предикторов на шаге t будем обозначать J_t. Вводится ограничение: $p_t = |J_t| \leq n_{max} = \min(m_0/2, n_0)$. Вначале $J_0 = \emptyset$.

Шаг t. Если $t > 1$, то делается попытка найти сначала столбец для *исключения* из J_{t-1}. Ищем столбец k, который дает минимальное приращение остаточной суммы квадратов (RSS), т.е. $k = \arg \max_l |V_l|$; $l \in J_{t-1}$, где $V_l = \tilde{a}^2_{l j_0} / |\tilde{a}_{ll}|$ (здесь $\tilde{a}_{ij}$ - текущие значения элементов матрицы A, a_{ij} - исходные значения). Столбец k исклю -

чается из подматрицы B ($J_t = J_{t-1} \setminus \{k\}$), если его F-значение исключения $FO_k < F_{out}$ - заданного уровня, где

$$FO_k = -(m_0 - p_{t-1} - 1)V_k / \tilde{a}_{j_0 j_0}, \quad p_{t-1} = |J_{t-1}|.$$

Если исключение невозможно ($FO_k \geq F_{out}$), делается попытка *включения* нового столбца в B. Ищем столбец $k = \arg \max_l V_l$; $l \in K \setminus J_{t-1}$, который обеспечивает максимальное уменьшение RSS. Далее, этот столбец включается в B ($J_t = J_{t-1} \cup \{k\}$), если а) его F-значение включения $FI_k > F_{in}$ - заданной константы, где

$$FI_k = (m_0 - p_{t-1} - 2)V_k / (\tilde{a}_{j_0 j_0} - V_k), \quad p_{t-1} = |J_{t-1}|,$$

и б) его толерантность $TO_k = \tilde{a}_{kk} / a_{kk} > TOL$ - заданного уровня; F_{in} и F_{out} контролируют значимость коэффициентов регрессии при соответствующих предикторах; параметр TOL ограничивает уровень корреляции между предикторами.

Если невозможно совершить ни шаг назад, ни шаг вперед, алгоритм завершает работу. При этом последний набор предикторов J_t, и результаты прогноза $\hat{y}_0(t')$, G_t, считаются оптимальными. Если уже $J_1 = \emptyset$, то прогноз $\hat{y}_0 = \bar{y}$, $G = (s_y\ 100)\%$.

После выбора очередного столбца x_k к текущей матрице $A = \hat{A} = \{\hat{a}_{jl}\}$ применяется оператор выметания по k-му столбцу $\hat{A} \to \tilde{A} = \{\tilde{a}_{jl}\}$:

$$\tilde{a}_{kk} = -1/\hat{a}_{kk}; \qquad \tilde{a}_{jk} = \hat{a}_{jk}/\hat{a}_{kk}; \qquad j, l \in K \setminus \{k\};$$

$$\tilde{a}_{kl} = \hat{a}_{kl}/\hat{a}_{kk}; \qquad \tilde{a}_{jl} = \hat{a}_{jl} - \hat{a}_{jk}\hat{a}_{kl}/\hat{a}_{kk}.$$

На каждой итерации вычисляется ожидаемая ошибка $G_t = (\tilde{a}_{j_0 j_0}/(m_0 - p_t - 1))^{1/2}\ 100\ \%$; коэффициенты регрессии $b_j = \tilde{a}_{j j_0}$, $j \in J$; $b_0 = \bar{y} - \sum_j b_j \bar{x}_j$, $j \in J_t$; значение прогноза $\hat{y}_0 = b_0 + \sum_j b_j x_{i_0 j}$, $j \in J_t$. Оценка G_t сильно

преуменьшает ошибку прогноза, можно предложить более точные оценки - классическую оценку дисперсии прогноза $D^2 = G_t^2((1/n_0) + d_{J_t}(\underline{x}_0, \bar{x}))$ и оценку ошибки по методу "кросс-проверки" (см. § 7):

$$G_{кп}^2 = \Big(\sum_{i \in I} (y_i - \hat{y}_i)^2/(1-h_i)^2\Big)/(n_0 - p_t - 1),$$

где $h_i = (1/n_0) + d_{J_t}(\underline{x}_i, \bar{x})$ - величина разбалансировки, а

$$d_{J_t}(\underline{x}_i, \bar{x}) = \sum_{l,j \in J_t} (-\tilde{a}_{lj})(x_{il} - \bar{x}_l)(x_{ij} - \bar{x}_j) -$$

расстояние Махалонобиса между строкой $\underline{x}_i$ и вектором средних $\bar{x} = (\bar{x}_{j_1}, \ldots, \bar{x}_{j_{n_0}})$ в пространстве столбцов $j \in J_t$;

$$\hat{y}_i = b_0 + \sum_j b_j x_{ij}; \quad j \in J_t, \; i \in I.$$

Можно порекомендовать следующие значения параметров $F_{in} = F_{out} = 3\text{-}4$, а $TOL = 0.1\text{-}0.2$.

§ 7. Алгоритм ZL-НЕП

В данном алгоритме используется непараметрическая оценка функции регрессии по m_0 ближайшим соседям с ядерными весами. Допускается нелинейность зависимости $g(\underline{x})$. Глобальные ядерные оценки регрессии восходят к работе Надарая [11], однако вычисление экспоненциальных весов для всех пар строк требует много времени; оценка же, реализованная здесь, ближе скорее к оценкам регрессии по методу ближайших соседей, предложенным Стоуном [35], хотя он употреблял веса более простого типа. Нам же нужны ядерные веса для более гибкого регулирования вклада ближайших соседей в прогноз (а также для уточнения их оптимального числа) посредством параметра сглаживания h. Для работы с алгоритмом требуется задать значения параметров: n_0, φ, k_g, k_p, t^*.

Прежде всего для каждого столбца $j \in K$ вычисляем матрицу W_j элементарных весов пар строк (i,k): $i,k \in I$, чтобы освободить процесс построения оптимального набора предикторов J от вычисления экспонент:

$$w_{ik}(j) = \begin{cases} \exp(-v_{ik}^2(j)), & \text{если } v_{ik}^2(j) \le \ln \varphi^{-1}, \\ 0 - \text{иначе}; \end{cases}$$

$$v_{ik}^2(j) = (x_{ij} - x_{kj})^2/(2h^2).$$

Здесь $0 < \varphi \le 1$ - параметр усечения малых весов (например, при $\varphi = 10^{-5}$ усекаются веса $w_{ik} < 10^{-5}$); $h > 0$ - параметр сглаживания ("ширина окна"). Итоговый прогноз будет находиться путем оптимизации критерия кросс-проверки по h; что же касается хода поиска J, то дефицит времени не позволяет оптимизировать тот же критерий при каждом J, поэтому критерий приходится вычислять при фиксированном h. Сильверман [33,с.86] указывает, что при радиально симметричном гауссовском ядре оптимальное значение h для нормально распределенных данных имеет вид:

$$h = \alpha(N,m_0)s_{cp}; \quad \alpha(N,m_0) = \left(\frac{4m_0^{-1}}{2N+1}\right)^{\frac{1}{N+4}};$$

$$s_{cp} = \frac{1}{N}\sum_j s_j,$$

$N = |K|$, $j \in K$. Так как у нас m_0 мало, то $\alpha(N,m_0)$ близка к 1 ($\alpha(N,m_0) < 1$ при $N > 1$) и $h = s_{cp}$ является более или менее подходящим выбором для h как при построении J, так и позже в качестве начального значения при оптимизации по h. Правильней было бы взять $s_{cp} = n_0^{-1}\sum_j s_j$,

$j \in J$, однако в этом случае для каждого J придется заново вычислять все экспоненты. В экспериментах использовано $h = s_{max} = \max\limits_{j} s_j$, $j \in K$, так как этот выбор исключает возможность возникновения ситуации, когда все веса усечены до нуля и для некоторого J невозможно вычислить критерий (а нормировать веса по максимальному весу нельзя из-за дефицита времени).

Построение множества J производится при помощи алгоритма SPA [10]. В ходе процесса генерируются k_g групп наборов столбцов из K по k_p наборов в каждой группе; k_g и k_p задаются пользователем. Рекомендуется $k_g, k_p \leq 10$ (практически неплохо работает выбор $k_g = k_p = 5$).

Пусть $|K| = N$. Каждый набор J должен содержать n_0 предикторов. Если $n_0 \geq N$, то $J \equiv K$. Если же $n_0 < N$, требуется отобрать n_0 из N столбцов. Когда $C_N^{n_0} \leq k_g k_p$, поиск J будет производиться путем полного перебора всех наборов из n_0 столбцов, среди которых определяется оптимальный по критерию D (см. ниже). Если же $C_N^{n_0} > k_g k_p$, в поиск включается алгоритм SPA.

Наборы первой группы J^1 в SPA наполняются случайно, в соответствии с вектором вероятностей выбора столбцов $P_0 = (p_{01}, \ldots, p_{0j}, \ldots, p_{0N})$, где $p_{0j} = 1/N$, $j \in K$. Датчик случайных чисел генерирует число $\eta : 0 \leq \eta \leq 1$, по которому определяется номер столбца $k = \min\{\nu | \sum\limits_{j=1}^{\nu} p_j \geq \eta , p_\nu > 0\}$. Тогда $J_1^1 = J_1^1 \cup \{k\}$, а вероятности пересчитываются $p_0' = (p_{01}', \ldots, p_{0j}', \ldots, p_{0N}')$,где $p_{0k}' = 0$, $p_{0j}' = p_{0j}/(1-p_{0k})$, $j \neq k$. Аналогично находятся и все остальные столбцы набора J_1^1, $|J_1^1| = n_0$, а затем - столбцы наборов $J_2^1, \ldots, J_{k_p}^1$, в результате чего формируется пер -

вая группа наборов $J^1 = \{J^1_1, \ldots, J^1_{k_p}\}$.

Ниже будет определен упрощенный критерий информативности набора J - $D(J)$. Вычисляем для каждого J^1_j значение $D(J^1_j)$, находим $D^1_{min} = D(J^1_{min}) = \min\limits_j D(J^1_j)$ и $D^1_{max} = D(J^1_{max}) = \max\limits_j D(J^1_j)$; $j = 1, \ldots, k_p$. Затем налагаем наказание на столбцы "наихудшего" набора J^1_{max}, т.е. от вектора p_0 переходим к вектору p_1: для $j \in K$, если $j \in J^1_{max}$, то $p_{1j} = p_{0j} - u$; а также поощряем столбцы "наилучшего" набора J^1_{min}, т.е. если $j \in J^1_{min}$, то $p_{1j} = p_{0j} + u$; $u = 1/(N \cdot k_g)$.

Далее, в соответствии с вектором p_1 по той же схеме генерируются k_p наборов второй группы $J^2 = \{J^2_1, \ldots, J^2_{k_p}\}$, для которой определяются J^2_{min}, J^2_{max}, D^2_{min}, D^2_{max} и аналогично применяется процедура наказания-поощрения. Таким путем получается k_g групп по k_p наборов в каждой. Оптимальный набор J: $D(J) = \min\limits_k D(J^k_{min})$, $k = 1, \ldots, k_g$.

Вариант с выбором столбцов посредством SPA назовем ZL-НСПА, а имя ZL-НЕП оставим за вариантом без отбора столбцов. Кроме того, можно предусмотреть варианты, в которых столбцы отбираются локально-параметрическими алгоритмами ZL-ПШ или ZL-СТ. Это резко сокращает количество операций, внося в то же время некоторую эклектичность - двойственность свойств в алгоритм. Назовем соответствующие варианты ZL-НПШ и ZL-НСТ.

В качестве критерия информативности D берется оценка по методу кросс-проверки. Пусть имеется набор J'. Определим матрицу весов W для него: $w_{ik} = \prod\limits_j w_{ik}(j)$, $j \in J'$; $i, k \in I$. Пусть $I_k = I \setminus \{k\}$; $L = \{k \mid \sum\limits_i w_{ik} > \varphi, i \in I_k\}$, $|L| = m_2$.

Обозначим $\tilde{w}_{ik} = w_{ik} / \sum_i w_{ik}$, $i \in I_k$, $k \in L$. Тогда критерий будет иметь вид:

$$D(J') = \frac{1}{m_2} \sum_{k \in L} (y_k - \sum_{i \in I_k} y_i \tilde{w}_{ik})^2 .$$

Можно показать, что $m_2 \neq 0$, если $\varphi = \varepsilon_0$, $h = s_{max}$.

Пусть построен набор $J = \{j_1, \ldots, j_{n_0}\}$ для подматрицы B; в качестве прогноза берем оценку:

$$\hat{y}_0(h) = \hat{E}(y | \underline{x}_0) = \frac{\int y \hat{p}(y, \underline{x}) dy}{\int \hat{p}(y, \underline{x}) dy} =$$

$$= (\sum_{i \in I} w_i y_i) / (\sum_{i \in I} w_i),$$

где $w_i = w_{i_0 i}$ - вес пары строк (i_0, i). Определим несимметричную матрицу весов $W(h)$ пар строк (i,k); $i,k \in I \cup \{i_0\}$, $i \neq k$:

$$w_{ki}(h) = \begin{cases} \exp(-v_{ki}^2), & \text{если } v_{ki}^2 \leq \ln \varphi^{-1}, \\ 0 - \text{иначе}; \end{cases}$$

$$r_{ik}^2 = \sum_{j \in J} (x_{ij} - x_{kj})^2; \quad v_{ki}^2 = (r_{ik}^2 - r_{min,k}^2)/(2h^2).$$

Здесь r_{ik} - локальное расстояние между строками i и k; $h > 0$ - параметр сглаживания (при $h \to 0$ алгоритм превращается в метод одного ближайшего соседа, при $h \to \infty$ - в локальное среднее); $r_{min,k} = \min_i r_{ik}$, $i \in I_k$, - расстояние от k-й строки до ее ближайшего соседа; $0 < \varphi \leq 1$ - параметр усечения малых весов.

Для выбора оптимального значения параметра h предусмотрена процедура минимизации по h критерия, определяемого процедурой кросс-проверки ("скользящего экзамена"), применительно к регрессии, называемой процедурой PRESS [16; 5,с.40]: а именно: вычисляем прогноз $\hat{y}_k(k)$ для каждого элемента отклика y_k, $k \in I$, по подмножеству строк $i \in I_k = I \setminus \{k\}$:

$$\hat{y}_k(k) = \hat{E}_k(y|\underline{x}_k) = \Big(\sum_{i \in I_k} y_i w_{ki}\Big) / \Big(\sum_{i \in I_k} w_{ki}\Big), \quad k \in I .$$

Благодаря нормировке $\forall k \in I, \exists i': \; w_{ki'} = 1$, поэтому $\sum_i w_{ki} \geq 1$, $i \in I_k$. Теперь определим оценку ожидаемой ошибки прогноза $D(h)$:

$$D(h) = \{\hat{E}(y-\hat{y})^2\}^{1/2} = \left\{\frac{1}{m_0} \sum_{k \in I} (y_k - \hat{y}_k(k))^2\right\}^{1/2} \cdot 100\,\% .$$

Оценка $D(h)$ используется в качестве критерия при поиске оптимального значения h. В ходе поиска для каждого значения h вычисляются $W(h)$, $\hat{y}_0(h)$ и $D(h)$. Оптимальным считается прогноз при $h = h^*$: $D(h^*) = \min\limits_h D(h)$. Оптимизация по h производится методом дихотомии на отрезке $h \in (0, s_{max})$. Можно предложить и некоторые другие варианты выбора отрезка, например, у Вагнера $h \in (0, r_{max})$ [3,с.159], где $r_{max} = \max\limits_i r_{i_0 i}$, $i \in I$. Однако эксперименты показали, что почти всегда $s_{max} < r_{max}$, а минимум как ожидаемой, так и фактической ошибки достигается, как правило, при $h < s_{max}$.

Учитывая трудоемкость вычисления $D(h)$, пользователь должен задать t^* - количество итераций, т.е. значений $D(h)$, после вычисления которых дихотомия завершает свою работу. Пусть минимальное значение $D(h^*)$ достигнуто при некотором

$h = h^*$. Тогда окончательный прогноз будет иметь вид $\hat{y}_0 = \hat{y}_0(h^*)$.

Критерий $D(h)$, к сожалению, очень чувствителен к ошибкам на дальних соседях, он склонен сильно преувеличивать ошибку прогноза; поэтому в качестве окончательного значения ожидаемой ошибки прогноза берется $G=G(h)=\underset{k\in I}{\text{med}}\,|y_k-\hat{y}_k(k)|\cdot 100\%$ при $h = h^*$, хотя эта оценка слегка преуменьшает ошибку. В заключение следует сказать, что оптимальные значения h и m_0 непосредственно связаны. Нужно задавать число ближайших соседей m_0 с некоторым избытком, после чего оптимизация по h определит их оптимальное число.

§ 8. Оценки времени и эксперименты

После завершения вычисления прогноза для пробела необходимо сделать обратную нормировку для прогноза:

$$\hat{x}'_{i_0 j_0} = \hat{y}_0 s'_{j_0} + \bar{x}'_{j_0}, \quad \text{если} \quad s'_{j_0} \geq \varepsilon .$$

Приведем оценки количества операций на один пробел. Все алгоритмы имеют общий член зависимости $O(mn + mm_0)$, связанный с построением матрицы C. При медианной нормировке он имеет вид $O(m^2 n)$. Кроме него есть и члены, по которым алгоритмы отличаются: ZL-СТ - $O(m_0 n_0 n)$, ZL-ПШ - $O(m_0 n^2)$, ZL-НСПА - $O(k_g k_p (n+m_0^2) + t^* m_0^2)$, ZET (по столбцам) - $O(m_0^2 n_0 t^*)$. Здесь $(m \times n)$ - размеры таблицы, $(m_0 \times n_0)$ - размеры подматрицы B, $k_g k_p$ - количество наборов, переби-раемых алгоритмом SPA, t^* - количество итераций по параметру h (ZL-НСПА) или α (ZET). Предполагается, что $m_0, n_0 < m$, $n_0 < n$. Когда n не слишком велико $n \sim (m_0 n_0 t^*)^{1/2}$, то по трудоемкости алгоритмы можно упорядочить: ZL-СТ $\prec$ ZL-ПШ $\prec$ $\prec$ ZET $\prec$ ZL-НЕП; если же n велико, то ZET $\prec$ ZL-ПШ. Опыт

показывает, что, как правило, все варианты алгоритма ZL-МЕП превосходят по времени остальные алгоритмы.

С целью сравнения алгоритмов были проведены следующие эксперименты. Таблица (60x12) заполнялась случайными числами из отрезка $[3,15]$. Каждый столбец x_j при этом приобретал равномерное распределение с матожиданием $E_x = 9$ и дисперсией $\sigma_x^2 = 12$. Тем самым исключалась возможность появления в таблице неконтролируемых зависимостей. На подматрице (21x6) строк, ближайших к первой строке, первый столбец $y = x_1$ заменялся линейной комбинацией следующих пяти столбцов: $y = 0.07x_2 - 0.2x_3 - 0.1x_4 + 0.18x_5 - 0.12x_6 + 8$, после чего к элементам столбца y добавлялись случайные ошибки ε_i с нормальным распределением $(E\varepsilon_i = 0)$, т.е. $y_i = \sum_{k=1}^{5} \alpha_k x_{i,k+1} + \alpha_0 + \varepsilon_i$, $i = 1,\ldots,21$. Дисперсия ошибок $\mathrm{Var}\,\varepsilon_i = \sigma^2$ выбиралась так, чтобы зависимость имела заданный множественный коэффициент корреляции $\rho^2 = (\mathrm{Var}\,y - \sigma^2)/\mathrm{Var}\,y$, где $\mathrm{Var}\,y = (\sum_{k=1}^{5} \alpha_k^2)\sigma_x^2 + \sigma^2$ - локальная дисперсия отклика. Для вычисления квадратической ошибки нужна еще глобальная дисперсия отклика $\mathrm{Var}'y$. Она находится как дисперсия смеси:

$$\mathrm{Var}'y = \frac{m_0}{m}\mathrm{Var}\,y + \frac{(m-m_0)}{m}\sigma_x^2 + \frac{m_0}{m}(Ey - E'y)^2 + \frac{(m-m_0)}{m}(E_x - E'y)^2;$$

$$Ey = (\sum_{k=1}^{5} \alpha_k)E_x; \quad E'y = (m_0 Ey + (m - m_0)E_x)/m;$$

$$E_x = 9, \quad \sigma_x^2 = 12, \quad m_0 = 20, \quad m = 60.$$

В таблице далее случайным образом удаляется заданный процент P наблюдений. Эксперимент заключается в редактировании элемента (1,1), т.е. y_1. Варьировались значения $\rho = 0.3$, 0.5, 0.7, 0.9, 1 и $P = 0$, 5, 10, 15, 20. Для каждого значения ρ (кроме $\rho = 1$) генерировалось по 10 векторов ошибок (соответственно 10 зависимостей) и, кроме того, 10 раз из таблицы случайным образом удалялся заданный процент наблюдений (при $P \neq 0$) для того, чтобы результаты не зависели от конкретной конфигурации пробелов. Таким образом, для каждой пары (ρ, P) генерировалось по 100, 10 либо 1 таблиц (всего 1681 таблица). Каждая таблица обрабатывалась пятью алгоритмами: ZET, ZL-СТ, ZL-ПШ, ZL-НЕП, ZL-НПШ. Для каждого прогноза $\hat{y}_1$ определялась фактическая относительная $|y_1 - \hat{y}_1|/|y_1|$ и квадратическая $|y_1 - \hat{y}_1|/(\mathrm{Var}'y)^{1/2}$ ошибки; эти ошибки затем усреднялись по 100 таблицам каждой пары (ρ, P).

В экспериментах использовались следующие значения параметров: $\varepsilon = 10^{-35}$, $\varepsilon_0 = 10^{-5}$, $m_0 = 20$, $n_0 = 5$; ZL-НЕП: $\varphi = 10^{-5}$, $\varphi_2 = 10^{-65}$, $t^* = 10$, $h = s_{max}$; ZL-ПШ: $F_{in} = F_{out} = 3.29$, $TOL = 0.2$; ZL-СТ: $g_0 = 1\%$, $\omega = 0.5$. Для тех, кто работает с алгоритмом ZET, сообщим, что оптимизация по параметру α велась на $[1,10]$ с шагом 1. Алгоритмам были сообщены оптимальные размеры подматриц, но строки и столбцы, входящие в эту подматрицу, они отыскивали самостоятельно, при этом поиску мешали пробелы.

Результаты экспериментов приведены в таблице. В каждой клетке таблицы, соответствующей паре значений (ρ, P), в левом столбце стоят средние относительные ошибки, в правом - средние квадратические.

Опыты показали, что при большом уровне корреляции ($\rho \geq 0.7$ и малом количестве пробелов ($P \leq 10\%$) наилучшие прогнозы дали локально-параметрические алгоритмы, а в остальных

Результаты экспериментов по сравнению алгоритмов на модельных данных

P	Алго-ритмы	Коэффициент множественной корреляции, ρ									
		0.3		0.5		0.7		0.9		1.0	
0	ZET	42.4	96.9	28.1	85.2	17.9	61.4	11.5	38.3	14.6	48.9
	ZL-НЕП	40.4	82.3	22.0	66.1	12.8	44.3	7.6	25.7	10,6	35.5
	ZL-НПШ	44.8	85.5	22.0	64.4	12.3	42.0	6.5	21.8	8.3	27.9
	ZL-ПШ	43.0	73.0	21.8	64.4	8.2	28.3	6.3	20.2	0.0	0.0
	ZL-СТ	47.5	90.4	27.6	73.5	10.1	34.7	6.7	21.2	1.4	4.7
5	ZET	42.4	97.2	26.7	80.9	18.3	62.6	11.1	37.3	14.9	49.8
	ZL-НЕП	41.5	83.9	22.5	67.7	12.9	44.5	7.9	26.5	10.9	36.5
	ZL-НПШ	32.1	66.0	23.7	71.2	6.3	22.5	6.3	21.0	10.9	36.4
	ZL-ПШ	32.8	60.7	24.8	69.7	9.8	33.1	8.8	27.9	6.9	23.1
	ZL-СТ	43.2	76.4	30.5	81.0	12.6	42.1	7.7	24.9	4.9	16.5
10	ZET	43.4	96.5	26.3	80.5	18.6	63.6	9.6	32.0	14.7	49.2
	ZL-НЕП	40.9	83.4	22.8	68.3	14.2	48.8	8.8	29.8	11.0	36.7
	ZL-НПШ	41.7	83.5	23.1	68.4	12.7	43.9	9.3	31.1	16.3	54.5
	ZL-ПШ	33.6	60.6	23.7	70.3	11.2	37.5	10.6	33.8	9.9	33.3
	ZL-СТ	40.4	66.4	25.9	73.4	13.9	46.0	9.1	28.8	9.8	33.0
15	ZET	44.6	95.7	25.2	79.8	17.7	60.7	16.5	54.8	15.1	50.7
	ZL-НЕП	40.8	83.4	22.9	69.1	14.8	50.9	9.2	30.8	11.5	38.5
	ZL-НПШ	39.8	81.7	25.5	75.4	13.5	46.1	11.1	37.0	11.7	39.2
	ZL-ПШ	43.0	86.2	26.3	76.6	12.5	42.9	13.8	45.5	10.3	34.4
	ZL-СТ	45.6	92.8	28.8	79.1	13.4	46.1	13.0	43.0	10.4	34.9
20	ZET	44.7	98.0	25.8	78.9	18.6	63.7	11.6	38.7	16.0	53.5
	ZL-НЕП	38.7	80.3	23.3	70.7	13.5	46.3	8.9	30.0	12.6	42.1
	ZL-НПШ	42.1	84.3	24.0	71.4	12.2	42.3	9.8	32.6	14.0	47.0
	ZL-ПШ	41.0	82.7	26.0	76.8	14.3	48.7	12.8	41.3	13.3	44.5
	ZL-СТ	45.4	90.2	29.1	81.8	16.4	56.0	14.8	47.6	9.4	31.4

случаях лучше работали ZL-НЕП и ZET, причем ZL-НЕП равномерно по всем парам (ρ, P) (а ZL-НПШ - почти по всем) дал меньшую среднюю ошибку, чем ZET. Это говорит о том, что ZL лучше оценивает множественные зависимости, чем ZET. Алгоритм ZL-НПШ несколько улучшил результаты ZL-НЕП при $\rho \geq 0.7$ и $P \leq 5\%$, ухудшив их в других случаях. Можно также отметить естественную тенденцию роста ошибки при уменьшении ρ и увеличении P, особенно четко проявляющуюся на локально-параметрических алгоритмах и в меньшей степени - на ZET и ZL-НЕП, для которых имеет место странный всплеск ошибки при $\rho = 1$.

По результатам экспериментов возникают вопросы. Сохраняется ли сравнительные достижения рассмотренных алгоритмов, если взять другие элементы для прогноза, например (2,1),(3,1),а еще лучше - если получить прогнозы для всех элементов первого столбца и усреднить ошибки этих прогнозов? Другой вариант: сгенерировать новую таблицу и снова предсказать элемент (1,1). Возможно, результаты будут иными. Интересно также было бы ввести в подматрицу дополнительные зависимости между предикторами, создав мультиколлинеарность в таблице. Это может выявить преимущества ZL-ПШ перед ZL-СТ. Неплохо бы попробовать нелинейные зависимости, а также генерировать ошибки из распределений, отличных от нормального, например, из равномерного или из распределений Лапласа.

§ 9. Рекомендации

Оценки трудоемкости показывают, что если нужен быстрый, но грубый прогноз, можно воспользоваться алгоритмом ZL-СТ. Несколько больше операций требует ZL-ПШ, но зато он способен довольно точно оценивать линейные зависимости, хотя и чувствителен к пробелам. В неблагоприятных ситуациях, когда нет оснований ожидать наличия в таблице линейных зависимостей или когда много пробелов в таблице, следует применять более гибкий ZL-НСПА при

небольших размерах подматрицы (<10-15 строк) и $k_g, k_p < 10$, помня о значительной трудоемкости этого алгоритма. При большом объеме редактируемых элементов (> 200)' придется обойтись вариантами ZL-НПШ или даже ZL-НЕП.

З а к л ю ч е н и е

Описанные эксперименты подтвердили целесообразность изменений, внесенных в базовую версию алгоритма ZET. В результате удалось построить ZL-семейство более эффективных алгоритмов заполнения пробелов и поиска ошибок в таблицах экспериментальных данных.

Авторы благодарят В.Н.Ёлкину и Т.П.Киприянову за полезные дискуссии в ходе разработки и сравнительных испытаний алгоритмов ZL-семейства.

Л и т е р а т у р а

1. АЙВАЗЯН С.А., ЕНЮКОВ И.С., МЕШАЛКИН Л.Д. Прикладная статистика: основы моделирования и первичная обработка данных. -М.: Финансы и статистика, 1983.- 471 с.

2. АФИФИ А., ЭЙЗЕН С. Статистический анализ. Подход с использованием ЭВМ.- М.: Мир, 1982.- 488 с.

3. ДЕВРОЙ Л., ДЬЁРФИ Л. Непараметрическое оценивание плотности. L_1-подход.-М.: Мир, 1988.- 408 с.

4. ДЖЕНРИЧ Р.И. Пошаговая регрессия// Статистические методы для ЭВМ.- М., 1986.- С.77-93.

5. ДРЕЙПЕР Н., СМИТ Г. Прикладной регрессионный анализ. Кн.2.: 2-е изд.- М.: Финансы и статистика,1987.- 351 с.

6. ЁЛКИНА В.Н., ЗАГОРУЙКО Н.Г., НОВОСЁЛОВ Ю.А. Математические методы агроинформатики.- Новосибирск,1987.- 202 с. (АН СССР. Сиб.отд-ние. Ин-т математики)

7. ЖАНАТАУОВ С.У.Методы прогностических переменных// Машинные методы обнаружения закономерностей.- Новосибирск,1981.- Вып.88: Вычислительные системы.- С.151-155.

8. ЗАГОРУЙКО Н.Г., ЁЛКИНА В.Н., ТИМЕРКАЕВ В.С. Алгоритм заполнения пропусков в эмпирических таблицах (алгоритм "ZET") //Вычислительные системы.- Новосибирск,1975.- Вып.61. Эмпирическое предсказание и распознавание образов.-С.3-27.

9. КЕНДАЛЛ М., СТЬЮАРТ А. Многомерный статистический анализ и временные ряды.- М.: Наука,1976.- 736 с.

10. ЛБОВ Г.С. Методы обработки разнотипных экспериментальных данных.- Новосибирск: Наука,1981.- 160 с.

11. НАДАРАЯ Е.А. Об оценке регрессии// Теория вероятностей и ее применения.-1964.-Т.9.- С.157-159.

12. РАО С.Р. Линейные статистические методы.- М.: Наука, 1968.- 548 с.

13. РАСТРИГИН Л.А., ПОНОМАРЕВ Ю.П. Экстраполяционные методы проектирования и управления.- М.: Машиностроение, 1986. - 120 с.

14. РАУДИС Ш., АЛЬФЕС М. Особенности решения задачи прогнозирования при ограниченном объеме выборки// Статистические проблемы управления. Вильнюс, 1986.- Вып.74.-С.132-145.

15. AFIFI A.A., ELASHOFF R.M. Missing observations in multivariate statistics// J.Amer. Statist. Assoc.-1966.-Vol.61. - P.595-604.

16. ALLEN D.M. Mean square error of prediction as a criterion for selecting variables// Technometrics.-1971.-Vol.13.- P.469-475.

17. BEALE E.M., LITTLE R.J. Missing values in multivariate analysis// J.Roy.Statist.Soc.B.-1975.-Vol.37.-P.129-145.

18. BUCK S.F. A method of estimation of missing values in multivariate data// J.Roy.Statist.Soc.B.-1960.- Vol.22.-P.202-206.

19. DEMPSTER A.P., LAIRD N.M., RUBIN D.B. Maximum likelihood from imcomplete data via the EM-algorithm// J.Roy.Statist. Soc.B.-1977.-Vol.39.-P.1-38.

20. DODGE Y. Amalysis of experiments with missing data. - New York: Wiley, 1985.- 499 p.

21. ENGELMAN L. An efficient algorithm for computing covariance matrices from data with missing values// Communs Statist.B.-1982.-Vol.11.-P.113-121.

22. FRANE G.M. Some simple procedures for handling miss - ing values in multivariate analysis// Psychometrika.- 1976. - Vol.41.- P.409-415.

23. GLASSER M. Linear regression analysis with missing observations among the independent variables//J.Amer.Statist.Assoc.-1964.-Vol.59.-P.834-844.

24. GLEASON T.C., STAELIN R. A proposal for handling missing data// Psychometrika.-1975.-Vol.40.-P.229-252.

25. HARTLEY H.O., HOCKING R.R. The analysis of incomplete data// Biometrics.-1971.-Vol.27.-P.783-808.

26. HILL R., ZIEMER R.F. Missing regressor values under conditions of multicollinearity// Communs Statist. Theory and Method.- 1983.- Vol.12.- P.2557-2573.

27. HOCKING R.R., MARX D.L. Estimation with incomplete data: an improved computational method and the analysis of nested data// Communs Statist.A.-1979.-Vol.8.-P.1151-1181.

28. HUSEBY J.R., SCHWERTMAN N.C., ALLEN D.M. Computation of the mean vector and dispersion matrix for incomplete multivariate data// Communs Statist.B.-1980.-Vol.9.-P.301-309.

29. LITTLE R.J., SCHLUSHTER M.D. Maximum likelihood estimation for mixed continuous and categorial data with missing values// Biometrika.-1985.-Vol.72.-P.497-512.

30. LITTLE R.J., RUBIN D.B. Statistical analysis with missing data.- New York: Wiley,1987.- 430 p.

31. LITTLE R.J., SMITH P.J. Editing and imputation for quantitative survey data// J.Amer.Statist.Assoc.-1987.-Vol.82.-P.58-68.

32. RUBIN D.B. Inference and missing data// Biometrika. - 1976.-Vol.63.-P.581-592.

33. SILVERMAN B.W. Density estimation for statistics and data analysis.- London: Chapman&Hall,1986.- 175 p.

34. SRIVASTAVA M.S. Multivariate data with missing observations// Communs Statist. Theory and Method.-1985. - Vol.14.-P.775-792.

35. STONE C.J. Consistent nonparametrical regression//Ann. Statist.-1977.-Vol.5.-P.595-645.

36. TABONY R.C. The estimation of missing values in highly correlated data// COMPSTAT-82. Proc.in Computational Sta - tist. 5-th Symp.- Wien,1982.-P.466-476.

37. TITTERINGTON D.M., JIANG J.M. Recursive estimation procedures for missing data problems// Biometrika. - 1983. - Vol.70.-P.613-624.

38. TITTERINGTON D.M., MILL G.M. Kernel-based density estimates from incomplete data// J.Roy.Statist.Soc.B.- 1983. - Vol.45.-P.258-267.

39. Walsh J.E. Computer-feasible method for handling incomplete data in regression analysis// J.of ACM.- 1961.-Vol.18. - P.201-211.

40. WILKS S.S. Moments and distributions of estimates of population from fragmentary samples// Ann.Math.Statist.-1932.-Vol.3.-P.163-195.

41. WANG S.G. A new iterative procedure for the missing-value problem// J.Comput.Math.-1984.-Vol.2.-P.234-238.

Поступила в ред.-изд.отд.
15 августа 1988 года