О ПРИБЛИЖЕННОМ РЕШЕНИИ ЗАДАЧ ВОССТАНОВЛЕНИЯ ЗАВИСИМОСТЕЙ С ПОМОЩЬЮ АЛГОРИТМОВ РАСПОЗНАВАНИЯ

Владимир Рязанов, Антон Щичко

Abstract: *Предлагается подход к восстановлению зависимостей, основанный на решении задач распознавания (классификации с учителем). Вычисление значения зависимой величины сводится к распознаванию интервала, которому она принадлежит. По данным обучения поставлена задача поиска оптимального разбиения области допустимых значений зависимой величины. Задача сформулирована в виде дискретной оптимизационной задачи. Вычисление значения оптимизируемой функции требует решения большого числа обучения распознаванию в режиме скользящего контроля. Для нескольких моделей распознавания типа вычисления оценок получены формулы быстрого переобучения при переходе от одной задачи распознавания к соседней. Приводятся результаты применения созданной модели и алгоритма восстановления значения зависимой величины для решения практических задач.*

Keywords: *восстановление зависимости, регрессия, классификация, алгоритм распознавания, прецедент, кусочно-постоянная функция, признак*

Введение

Многие задачи анализа прецедентных данных часто задаются в следующем стандартном виде. Дана выборка $\{z_i, \mathbf{x}_i\}, i = 1,2,...,m$, где $\mathbf{x}_i = (x_{i1}, x_{i2},..., x_{in})$ - признаковое описание объекта, $z_i \in R$, $x_{ij} \in M_j$ (M_j - известное множество допустимых значений признака № j). Вектор значений $\mathbf{x}_i$ задает исходные характеристики объекта, которые можно вычислить или измерить. Поэтому мы будем называть его вектором значений независимых параметров. Предполагается, что величина z_i , известная для объектов обучающей выборки, является скрытой от наблюдения главной характеристикой объекта, которая может быть вычислена по вектору $\mathbf{x}_i$, т.е. $z_i = f(\mathbf{x}_i)$. Будем ее называть зависимой величиной. Требуется для произвольного нового $\mathbf{x} = (x_1, x_2,..., x_n)$, $x_j \in M_j$, вычислить $z = f(\mathbf{x})$, $z \in R$, используя данные обучающей выборки. В статистической постановке данная задача известна как задача восстановления регрессии - функции условного математического ожидания, при предположении существования условной плотности $p(z \mid \mathbf{x})$. В данной статье не будут использоваться какие-либо вероятностные модели и предположения.

В настоящее время существуют различные параметрические и непараметрические подходы к восстановлению зависимостей при $x_i \in R, i = 1,2,..., n$.

Параметрический подход [Дрейпер, Смит, 2007] предполагает наличие функциональной зависимости от некоторых параметров $\boldsymbol{\omega}$, причем вид зависимости известен:

- линейная зависимость - $f(\boldsymbol{\omega}, \mathbf{x}) = \sum_{j=1}^{n} \omega_j x_j + \omega_0$;

- полиномиальная зависимость степени γ - $f(\boldsymbol{\omega},\mathbf{x}) = \sum_{p_1=0}^{\gamma_1} ... \sum_{p_n=0}^{\gamma_n} \omega_{p_1 ... p_n} x_1^{p_1} ... x_n^{p_n}$, $\gamma = \sum_{i=1}^{n} \gamma_i$;
- криволинейная зависимость - $f(\boldsymbol{\omega},\mathbf{x}) = \sum_{j=1}^{k} \omega_j \phi_j (x_1,...,x_n) + \omega_0$, $\phi_1,...,\phi_k$ - преобразования $R^n \to R$;
- логистическая зависимость: $f(\boldsymbol{\omega},\mathbf{x}) = \dfrac{1}{1+\exp(-z)}$, $z = \sum_{j=1}^{n} \omega_j x_j + \omega_0$.

В непараметрическом подходе [Хардле, 1993] характеристика z для $\mathbf{x}$ определяется как $z = \dfrac{\sum_{i=1}^{m} \omega_i(\mathbf{x}) z_i}{\sum_{i=1}^{m} \omega_i(\mathbf{x})}$, где $\omega_i(\mathbf{x}) = K\left(\dfrac{\rho(\mathbf{x},\mathbf{x}_i)}{h}\right)$, $i = 1,2,...,m$, K - ядерная функция, h - ширина окна.

Широко известны методы регрессии основанные на опорных векторах [Collobert, Bengio, 2001], которые могут рассматриваться как разновидность криволинейных регрессий. В работе [Jing-Rung Yu, Gwo-Hshiung Tzeng, Han-Lin Li 2001] был предложен общий метод построения нечеткой кусочной регрессии, где точки разбиения для значений зависимой величины вычисляются одновременно как решения смешанной задачи математического программирования. Существуют и другие близкие по сути подходы.

Отметим главные ограничения данных подходов. Параметрические подходы требуют априорного знания аналитического вида функций. Наличие разнотипных признаков (вещественных, номинальных, бинарных, порядковых, и т.п.) требует привлечения дополнительных средств описания объектов в единой шкале. Непараметрические методы широко используют частотные оценки, функции расстояний, что может быть весьма приближенным и практически затруднительным для выборок малой длины, при большом числе независимых параметров, различной их информативности и разнотипности, при наличии шумовых признаков.

В настоящей статье предлагается подход, не использующий априорные вероятностные предположения и основанный на теории распознавания (классификации с учителем). Задача решается следующим образом. Каждое разбиение отрезка $[a,b]$, $a = \min_{i=1,2,...,m} z_i$, $b = \max_{i=1,2,...,m} z_i$ на конечное число отрезков порождает некоторую задачу распознавания по прецедентам и определяется значениями некоторого вектора параметров $\mathbf{y}$ с конечным числом возможных значений. Для стандартной задачи распознавания по прецедентам $\mathbf{Z}(\mathbf{y})$ решается задача обучения и находится алгоритм классификации $A(\mathbf{y})$, качество которого оценивается функцией $F(A(\mathbf{y}))$ в режиме скользящего контроля (leave-one-out procedure). Нахождение оптимального числа точек разбиения и самих точек сводится в минимизации $F(A(\mathbf{y}))$. С каждым классом связывается некоторое значение зависимой величины, оцениваемое по объектам обучения данного класса (выборочное среднее, среднее от пары точек, и т.д.). Вычисление функции $F(A(\mathbf{y}))$ в режиме скользящего контроля требует многократного обучения для соседних задач. Для некоторых алгоритмов типа вычисления оценок получены эффективные формулы переобучения алгоритмов при переходе от одной стандартной задачи распознавания к соседней ей. Вычисление значения зависимой величины для некоторого $\mathbf{x}$ сводится к распознаванию класса объекта относительно найденного оптимального разбиения.

Далее, без ограничения общности, будем считать, что значения z_i различны, а объекты обучающей выборки упорядочены по возрастанию значений z_i, т.е. $z_i < z_{i+1}, i = 1,2,...,m-1$.

Постановка задачи восстановления зависимостей в классе кусочно-постоянных функций

Пусть фиксирован некоторый вектор $\mathbf{y} = (y_1,...,y_{l-1})$, $y_1 < y_2 < ... < y_{l-1}$, $z_2 \le y_i \le z_{m-2}, i = 1,2,...,l-1,$ $y_i < y_{i+1}, i = 1,2,...,l-2$, $y_i \in \{z_j \mid j = 1,2,...,m\}$. Набор $\mathbf{y}$ определяет разбиение вещественной оси на l множеств $I_1 = (-\infty, y_1], I_2 = (y_1, y_2],..., I_l = (y_{l-1}, +\infty)$ и разбиение множества допустимых признаковых описаний объектов на l классов $K_j = \{\mathbf{x} : z = f(\mathbf{x}) \in I_j\}, j = 1,2,...,l$. Множества K_j задаются в виде $\widetilde{K}_j = \{\mathbf{x}_i : z_i \in I_j, i = 1,2,...,m\}, j = 1,2,...,l$.

Пусть фиксирован некоторый алгоритм классификации $A^{\mathbf{y}}$ относительно классов $K_j, j = 1,2,...,l$, заданных вектором $\mathbf{y}$. Через $\widetilde{A}_i^{\mathbf{y}}$ обозначим алгоритм распознавания, соответствующий вектору $\mathbf{y}$, после удаления из обучающей выборки объекта $\mathbf{x}_i$. Определим функцию $f(\mathbf{x}|A^{\mathbf{y}})$ следующим образом:

$$f(\mathbf{x}|A^{\mathbf{y}}) = \begin{cases} (y_1 + z_1)/2, & \mathbf{x} \xrightarrow{A^{\mathbf{y}}} K_1, \\ (y_j + y_{j-1})/2, & \mathbf{x} \xrightarrow{A^{\mathbf{y}}} K_j, \quad j = 2,3,...,l-1, \\ (z_m + y_{l-1})/2, & \mathbf{x} \xrightarrow{A^{\mathbf{y}}} K_l, \end{cases} \qquad (1)$$

если объект $\mathbf{x}$ отнесен алгоритмом $A^{\mathbf{y}}$ в некоторый класс. Данный результат классификации будем обозначать как $A^{\mathbf{y}}(\mathbf{x}) = j$. Соответственно, $\widetilde{A}_i^{\mathbf{y}}(\mathbf{x}_i) = j$ будет обозначать результат распознавания (номер класса) алгоритмом $\widetilde{A}_i^{\mathbf{y}}$ объекта $\mathbf{x}_i$.

Будем использовать обозначение $|I_t| = |\{z_i : z_i \in I_t\}|, t = 1,2,...l$.

Рассмотрим следующую оптимизационную задачу.

$$F(y_1, y_2,..., y_{l-1}) = \sum_{z_i : z_i \le y_1} \left| z_i - f(\mathbf{x}_i|\widetilde{A}_i^{\mathbf{y}}) \right| + \sum_{i=1}^{l-2} \sum_{z_i : y_i < z_i \le y_{i+1}} \left| z_i - f(\mathbf{x}_i|\widetilde{A}_i^{\mathbf{y}}) \right| + \sum_{z_i : y_{l-1} \le z_i} \left| z_i - f(\mathbf{x}_i|\widetilde{A}_i^{\mathbf{y}}) \right| \to \min,$$

$$z_3 \le y_i \le z_{m-2}, i = 1,2,...,l-1, \ y_i < y_{i+1}, i = 1,2,...,l-2, \ |I_1| \ge 3, \ |I_t| \ge 2, t = 2,3,...l. \qquad (2)$$

Ограничения на $|I_t| = |\{z_i : z_i \in I_t\}|, t = 1,2,...l$, связаны с видом функции (1), поскольку значение зависимой величины вычисляется по двум точкам выборки, и процедурой скользящего контроля.

Оптимальное решение $y_1, y_2,..., y_{l-1}$ и определяет точки разбиения значений зависимой величины и оптимальную стандартную задачу распознавания.

После нахождения оптимальных значений $y_1, y_2,..., y_{l-1}$, вычисление значений зависимой величины осуществляется следующим образом.

1. Решается задача классификации объекта $\mathbf{x}$ алгоритмом A^{y}.
2. Вычисление $\mathbf{z} = f(\mathbf{x}) \equiv f(\mathbf{x} | A^{\mathrm{y}})$ проводится по формуле (1).

Отметим, что при вычислении по формуле (1) значения зависимой величины мы использовали среднее по граничным точкам. Здесь могут быть использованы и другие способы ее оценки (выборочное среднее по классу, медиана, и т.п.). При классификации $\mathbf{x}$ мы считаем, что результат «отказ от классификации» не существует. В данном подходе может быть использована произвольная модель распознавания. Основная вычислительная проблема здесь связана с вычислением $F(y_1, y_2, ..., y_{l-1})$ в режиме скользящего контроля, когда требуется быстрое переобучение алгоритма при исключении некоторого объекта из обучающей выборки. Данное переобучение осуществляется быстро в некоторых моделях распознавания типа вычисления оценок. Далее мы рассмотрим общую схему оптимизации и используемые алгоритмы распознавания. Отметим, что вычисляемая в итоге функция является кусочно-постоянной.

Алгоритм поиска оптимальной кусочно-постоянной функции по обучающей выборке

Рассмотрим задачу (2), где функция $f(\mathbf{x})$ (и соответствующий алгоритм распознавания A), задаются текущими значениями параметров $y_1, y_2, ..., y_{l-1}$. Объекты обучения упорядочены по возрастанию z_i и имеет место $y_t = z_{i_t}$. Рассматривалась схема локальной оптимизации $F(y_1, y_2, ..., y_{l-1})$. Точки $\mathbf{y}^* = (y_1^*, y_2^*, ..., y_{l-1}^*)$, $y_j^* \in \{z_{i_j - 1}, z_{i_j + 1}\}$, $y_t^* = y_t, t \neq j$, $j = 1,2,...,l-1$, назовем соседними для $\mathbf{y} = (y_1, y_2, ..., y_{l-1})$. Начиная с произвольного допустимого $\mathbf{y}^{(0)} = (y_1^{(0)}, y_2^{(0)}, ..., y_{l-1}^{(0)})$ осуществляем просмотр всех не более чем $2(l-1)$ соседних допустимых (т.е. удовлетворяющих ограничениям (2)) точек. Обозначим через $\mathbf{y}' = (y_1', y_2', ..., y_{l-1}')$ произвольную соседнюю допустимую точку для $\mathbf{y}^{(0)} = (y_1^{(0)}, y_2^{(0)}, ..., y_{l-1}^{(0)})$. В качестве точки минимума $F(y_1, y_2, ..., y_{l-1})$ в окрестности $\mathbf{y}^{(0)}$ принимаем произвольную соседнюю допустимую точку $\mathbf{y}^{(1)} = (y_1^{(1)}, y_2^{(1)}, ..., y_{l-1}^{(1)})$, для которой $F(y_1^{(1)}, y_2^{(1)}, ..., y_{l-1}^{(1)}) \le F(y_1', y_2', ..., y_{l-1}')$, $F(y_1^{(1)}, y_2^{(1)}, ..., y_{l-1}^{(1)}) < F(y_1^{(0)}, y_2^{(0)}, ..., y_{l-1}^{(0)})$. Осуществляется переход в точку $\mathbf{y}^{(1)}$ и процесс повторяется. Алгоритм локальной оптимизации заканчивает работу на шаге t, если выполняется условие $F(y_1^{(t-1)}, y_2^{(t-1)}, ..., y_{l-1}^{(t-1)}) \le F(y_1', y_2', ..., y_{l-1}')$ для всех соседних вектору $(y_1^{(t-1)}, y_2^{(t-1)}, ..., y_{l-1}^{(t-1)})$ допустимых точек $(y_1', y_2', ..., y_{l-1}')$. Конечность локальной оптимизации следует из конечности допустимых значений $F(y_1, y_2, ..., y_{l-1})$. Отметим, что алгоритм динамического программирования [Михалевич, 1965] в здесь не применим, поскольку слагаемые в (2) зависят не только от значений одного-двух переменных y_i.

Модификация алгоритма вычисления оценок (АВО). Восстановление зависимостей с использованием модели распознавания АВО.

Опишем принцип работы алгоритмов вычисления оценок [Журавлев, Никифоров, 1971]. В выше приведенных обозначениях $z_i, z \in \{1,2,...,l\}$. Пусть некоторое множество X допустимых объектов $\mathbf{x}$

имеет вид $X = \bigcup_{j=1}^{l} K_j, K_\nu \cap K_\mu = \varnothing,\ \nu \neq \mu$. Дана обучающая выборка $\{z_t, \mathbf{x}_t, t = 1,2,...,m\}$, где $z_t = j$, если $\mathbf{x}_t \in K_j$. Для простоты считаем, что $\mathbf{x} = (x_1, x_2, ..., x_n) \in R^n$, а обучающая выборка содержит представителей всех классов. Обозначим $\widetilde{K}_j = \{\mathbf{x}_i : \mathbf{x}_i \in K_j, i = 1,2,...,m\}$. Пусть фиксирована система опорных множеств $\Omega_A = \{\Omega\}$, $\Omega \subseteq \{1,2,...,n\}$ алгоритма A. Опорное множество Ω задает некоторое подмножество признаков. Близость распознаваемого объекта $\mathbf{x}$ к некоторому объекту обучения $\mathbf{x}_t$ по опорному множеству Ω определяется как

$$\mathrm{B}_\Omega(\mathbf{x}, \mathbf{x}_t) = \begin{cases} 1, & |x_i - x_{ti}| \leq \varepsilon_i, \forall i \in \Omega, \\ 0, & \text{иначе.} \end{cases} \tag{3}$$

В [Журавлев, Никифоров, 1971] присутствуют числовые параметры $\varepsilon_i, i = 1,2,...,n$, задаваемые пользователем или вычисляемые как, например, $\varepsilon_i = \frac{2}{m(m-1)} \sum_{\alpha,\beta=1,\alpha<\beta}^{m} |x_{\alpha i} - x_{\beta i}|$. Для объекта $\mathbf{x}$ вычисляется оценка $\Gamma_j(\mathbf{x})$ за класс $K_j, j = 1,2,...,l$:

$$\Gamma_j(\mathbf{x}) = \frac{1}{|\widetilde{K}_j|} \sum_{\mathbf{x}_i \in \widetilde{K}_j} \sum_{\Omega \in \Omega_A} B_\Omega(\mathbf{x}, \mathbf{x}_i). \tag{4}$$

Оценка $\Gamma_j(\mathbf{x})$ характеризует эвристическую степень близости объекта $\mathbf{x}$ к классу K_j. Далее применяется решающее правило в пространстве оценок: объект $\mathbf{x}$ относится алгоритмом A в класс K_j, если $\Gamma_j(\mathbf{x}) > \Gamma_i(\mathbf{x}), \forall i \neq j$. В противном случае выбор класса происходит из классов с максимальными оценками случайно. Обычно используют в качестве системы опорных множеств $\Omega_A = \{\Omega : |\Omega| = k\}$, где $0 \leq k \leq n$, k - целое, либо Ω_A есть система всех подмножеств множества $\{1,2,...,n\}$. Параметр k является внешним, мы использовали $k = \left[\frac{n}{3}\right]$. В [Журавлев, Никифоров, 1971] доказано, что при первом способе выбора $\Gamma_j(\mathbf{x}) = \frac{1}{|\widetilde{K}_j|} \sum_{\mathbf{x}_i \in \widetilde{K}_j} C^k_{d(\mathbf{x},\mathbf{x}_i)}$, а во втором случае $\Gamma_j(\mathbf{x}) = \frac{1}{|\widetilde{K}_j|} \sum_{\mathbf{x}_i \in \widetilde{K}_j} (2^{d(\mathbf{x},\mathbf{x}_i)} - 1)$. Здесь $d(\mathbf{x}, \mathbf{x}_i) = \left| \{ j : |x_j - x_{ij}| \leq \varepsilon_j, j = 1,2,...,n \} \right|$.

В настоящей работе использовалась следующая модификация функции близости (3) и формулы (4). Пусть $\mathbf{x}_\alpha, \mathbf{x}_\beta \in \widetilde{K}_j$, тогда определим функцию близости $\widetilde{\mathrm{B}}_\Omega(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta)$ объекта $\mathbf{x}$ к паре $\mathbf{x}_\alpha, \mathbf{x}_\beta$, и его оценку $\widetilde{\Gamma}_j(\mathbf{x})$ за класс K_j следующими выражениями.

$$\widetilde{\mathrm{B}}_\Omega(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta) = \begin{cases} 1, & (x_{\alpha i} \leq x_i \leq x_{\beta i}) \vee (x_{\beta i} \leq x_i \leq x_{\alpha i}), \forall i \in \Omega, \\ 0, & \text{иначе.} \end{cases} \tag{5}$$

$$\widetilde{\Gamma}_j(\mathbf{x}) = \frac{2}{|\widetilde{K}_j|(|\widetilde{K}_j| - 1)} \sum_{\mathbf{x}_\alpha, \mathbf{x}_\beta \in \widetilde{K}_j, \alpha<\beta} \left(\sum_{\Omega \in \Omega_A} \widetilde{\mathrm{B}}_\Omega(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta) \right). \tag{6}$$

Можно показать, что здесь также справедливы аналогичные эффективные формулы для вычисления оценок:

$\widetilde{\Gamma}_j(\mathbf{x}) = \frac{2}{\left|\widetilde{K}_j\right|\left(\left|\widetilde{K}_j\right|-1\right)} \sum_{\mathbf{x}_\alpha,\mathbf{x}_\beta \in \widetilde{K}_j, \alpha<\beta} \mathrm{C}^k_{d(\mathbf{x}_\alpha,\mathbf{x},\mathbf{x}_\beta)}$, $\widetilde{\Gamma}_j(\mathbf{x}) = \frac{2}{\left|\widetilde{K}_j\right|\left(\left|\widetilde{K}_j\right|-1\right)} \sum_{\mathbf{x}_\alpha,\mathbf{x}_\beta \in \widetilde{K}_j, \alpha<\beta} (2^{d(\mathbf{x}_\alpha,\mathbf{x},\mathbf{x}_\beta)}-1)$, где $d(\mathbf{x}_\alpha,\mathbf{x},\mathbf{x}_\beta) = \left|\{ j : (x_{\alpha i} \le x_i \le x_{\beta i}) \vee (x_{\beta i} \le x_i \le x_{\alpha i}), j = 1,2,...,n\}\right|$. После вычисления оценок $\widetilde{\Gamma}_j(\mathbf{x}), j = 1,2,...,l,$ используется приведенное ранее решающее правило. Отметим, что в данном случае не используется метрика в R и параметры $\varepsilon_i, i = 1,2,...,n$. Признаки могут быть порядковыми.

Настоящий алгоритм не содержит параметров, требующих настройки при обучении.

Обозначим $m_j = \left|\widetilde{K}_j\right|, j = 1,2,...,l$. Рассмотрим задачу оптимизации (2), когда в качестве базовой модели распознавания используется модифицированная модель вычисления оценок.

При решении задачи оптимизации (2) оценки $\widetilde{\Gamma}_j(\mathbf{x})$ легко пересчитываются в режиме скользящего контроля.

Действительно, вычислим матрицы, $\mathbf{D}^1 = \left\| D^1_{\alpha\gamma\beta} \right\|_{m\times m\times m}$, $D^1_{\alpha\gamma\beta} = \mathrm{C}^k_{d(\mathbf{x}_\alpha,\mathbf{x}_\gamma,\mathbf{x}_\beta)}$, $\mathbf{D}^2 = \left\| D^2_{\alpha\gamma\beta} \right\|_{m\times m\times m}$, $D^2_{\alpha\gamma\beta} = 2^{d(\mathbf{x}_\alpha,\mathbf{x}_\gamma,\mathbf{x}_\beta)} - 1$. Пусть $\mathbf{x}_t$ - произвольный объект выборки, имеется текущее разбиение на классы $K_1, K_2,..., K_l$ и $\mathbf{x}_t \in K_i$. Тогда $\widetilde{\Gamma}_i(\mathbf{x}_t) = \frac{2}{(m_i-1)(m_i-2)} \sum_{\substack{\alpha<\beta:\alpha,\beta\neq t \\ \mathbf{x}_\alpha,\mathbf{x}_\beta \in K_i,}} D^h_{\alpha t\beta}$, $\widetilde{\Gamma}_j(\mathbf{x}_t) = \frac{2}{m_j(m_j-1)} \sum_{\substack{\alpha<\beta: \\ \mathbf{x}_\alpha,\mathbf{x}_\beta \in K_j,}} D^h_{\alpha t\beta}, j \neq i$. Здесь $h \in \{1,2\}$ и для простоты записи мы его опускаем далее в выражениях $\widetilde{\Gamma}_j(\mathbf{x}_t)$. Рассмотрим пересчет оценок $\widetilde{\Gamma}_j(\mathbf{x}_t), j = 1,2,...,l$, при пересчете функционала в соседней точке произвольного шага алгоритма оптимизации. При этом граница между некоторой парой классов меняется в результате переноса некоторого объекта $\mathbf{x}_\tau$ из одного класса в соседний класс. Обозначим «новые» классы $K^*_1, K^*_2,..., K^*_l$, а оценки для $\mathbf{x}_t$ через $\widetilde{\Gamma}^*_j(\mathbf{x}_t), j = 1,2,...,l$.

Возможны следующие четыре варианта:

1. $K^*_i = K_i \cup \{\mathbf{x}_\tau\}$, $\mathbf{x}_\tau \in K_u, u \neq i$,

$K^*_u = K_u \setminus \{\mathbf{x}_\tau\}$,

$K^*_j = K_j, j \neq i, u$.

2. $K^*_u = K_u \setminus \{\mathbf{x}_\tau\}$,

$K^*_v = K_v \cup \{\mathbf{x}_\tau\}$,

$K^*_j = K_j, j \neq u, v, \quad u, v \neq i$.

3. $K_i^* = K_i \setminus \{\mathbf{x}_\tau\}, \tau \neq t,$

$K_u^* = K_u \cup \{\mathbf{x}_\tau\},$

$K_j^* = K_j, j \neq i, u$.

4. $K_i^* = K_i \setminus \{\mathbf{x}_t\},$

$K_u^* = K_u \cup \{\mathbf{x}_t\},$

$K_j^* = K_j, j \neq i, u$.

Тогда оценки $\widetilde{\Gamma}_j^*(\mathbf{x}_t), j = 1,2,...,l$, пересчитываются следующим образом:

1. $$\widetilde{\Gamma}_i^*(\mathbf{x}_t) = \frac{2}{m_i(m_i-1)}\left(\frac{(m_i-1)(m_i-2)}{2}\widetilde{\Gamma}_i(\mathbf{x}_t) + \sum_{\substack{\alpha:\alpha\neq t \\ \mathbf{x}_\alpha\in K_i, \\ \mathbf{x}_\tau\in K_u, u\neq i}} D_{\alpha\tau}^h\right),$$

$$\widetilde{\Gamma}_u^*(\mathbf{x}_t) = \frac{2}{(m_u-1)(m_u-2)}\left(\frac{m_u(m_u-1)}{2}\widetilde{\Gamma}_u(\mathbf{x}_t) - \sum_{\substack{\alpha:\mathbf{x}_\alpha\in K_u, \\ \mathbf{x}_\tau\in K_u, u\neq i}} D_{\alpha\tau}^h\right),$$

$\widetilde{\Gamma}_j^*(\mathbf{x}_t) = \widetilde{\Gamma}_j(\mathbf{x}_t), j \neq i, u$.

2. $$\widetilde{\Gamma}_u^*(\mathbf{x}_t) = \frac{2}{(m_u-1)(m_u-2)}\left(\frac{m_u(m_u-1)}{2}\widetilde{\Gamma}_u(\mathbf{x}_t) - \sum_{\substack{\alpha:\mathbf{x}_\alpha\in K_u, \\ \mathbf{x}_\tau\in K_u, u\neq i}} D_{\alpha\tau}^h\right),$$

$$\widetilde{\Gamma}_v^*(\mathbf{x}_t) = \frac{2}{(m_v+1)m_v}\left(\frac{m_v(m_v-1)}{2}\widetilde{\Gamma}_v(\mathbf{x}_t) + \sum_{\substack{\alpha:\mathbf{x}_\alpha\in K_v, \\ \mathbf{x}_\tau\in K_u, u\neq i}} D_{\alpha\tau}^h\right) u, v \neq i,$$

$\widetilde{\Gamma}_j^*(\mathbf{x}_t) = \widetilde{\Gamma}_j(\mathbf{x}_t), j \neq u, v$.

3. $$\widetilde{\Gamma}_i^*(\mathbf{x}_t) = \frac{2}{(m_i-2)(m_i-3)}\left(\frac{(m_i-1)(m_i-2)}{2}\widetilde{\Gamma}_i(\mathbf{x}_t) - \sum_{\substack{\alpha:\mathbf{x}_\alpha\in K_i, \\ \mathbf{x}_\tau\in K_i, \tau\neq t}} D_{\alpha\tau}^h\right),$$

$$\widetilde{\Gamma}_u^*(\mathbf{x}_t) = \frac{2}{(m_u+1)m_u}\left(\frac{m_u(m_u-1)}{2}\widetilde{\Gamma}_u(\mathbf{x}_t) + \sum_{\substack{\alpha:\mathbf{x}_\alpha\in K_u, \\ \mathbf{x}_\tau\in K_i, \tau\neq t}} D_{\alpha\tau}^h\right),$$

$\widetilde{\Gamma}_j^*(\mathbf{x}_t) = \widetilde{\Gamma}_j(\mathbf{x}_t), j \neq i, u$.

4. $\widetilde{\Gamma}_i^*(\mathbf{x}_t) = \widetilde{\Gamma}_i(\mathbf{x}_t),$

$$\widetilde{\Gamma}_u^*(\mathbf{x}_t) = \frac{2}{(m_u+1)m_u}\left(\frac{m_u(m_u-1)}{2}\widetilde{\Gamma}_u(\mathbf{x}_t) + \sum_{\substack{\alpha<\beta:\\ \mathbf{x}_\alpha,\mathbf{x}_\beta\in K_u,}} D_{\alpha\beta}^h\right), u \neq i,$$

$$\widetilde{\Gamma}_j^*(\mathbf{x}_t) = \widetilde{\Gamma}_j(\mathbf{x}_t),\ j \neq i, u.$$

Таким образом, вычисление функционала в соседней точке общего алгоритма оптимизации осуществляется эффективно.

Восстановление зависимостей с использованием модели распознавания, основанной на голосовании по системам логических закономерностей классов

Алгоритмы распознавания, основанные на голосовании по системам логических закономерностей классов, описаны в [Рязанов, 2007; Ковшов, Моисеев, Рязанов, 2008] и работают следующим образом. Рассматривается множество элементарных предикатов, зависящих от числовых параметров $c_j^1, c_j^2, j = 1,2,...,n:\ P_j^{c_j^1,c_j^2}(x) = \begin{cases} 1, & c_j^1 \leq x \leq c_j^2, \\ 0, & \text{иначе} \end{cases}$. Пусть $\Omega \subseteq \{1,2,...,n\}$.

Определение [Рязанов, 2007] . Предикат $P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}) = \underset{j\in\Omega}{\&} P_j^{c_j^1,c_j^2}(x_j)$ называется логической закономерностью (ЛЗ) класса $K_\lambda, \lambda = 1,2,...,l,$ если

1. $\exists \mathbf{x}_t \in \widetilde{K}_\lambda : P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}_t) = 1,$
2. $\exists \mathbf{x}_t \in \widetilde{K}_\mu, \mu = 1,2,...,l, \mu \neq \lambda : P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}_t) = 0,$

$\Phi(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})) = \underset{\{P^{\Omega',\mathbf{c}^{1*},\mathbf{c}^{2*}}(\mathbf{x})\}}{extr} \Phi(P^{\Omega',\mathbf{c}^{1*},\mathbf{c}^{2*}}(\mathbf{x}))$, где Φ - критерий качества предиката.

Будем далее использовать стандартный критерий F качества класса $K_\lambda, \lambda = 1,2,...,l$: $F(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})) = \left|\{\mathbf{x}_i : \mathbf{x}_i \in \widetilde{K}_\lambda, P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}_i) = 1, i = 1,2,...,m\}\right|$ и понятия эквивалентных ЛЗ (принимающих равные значения на объектах обучающей выборки) и интервалов $N(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$ ЛЗ $P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})$. Пусть $\mathrm{X}(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}) = \{\mathbf{x}_t \in K_\lambda : P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}_t) = 1\}$. Будем называть $P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})$ минимальной, если не существует ей эквивалентная ЛЗ $P^{\Omega_0,\mathbf{c}^3,\mathbf{c}^4}(\mathbf{x})$, для которой $N(P^{\Omega_0,\mathbf{c}^3,\mathbf{c}^4}(\mathbf{x})) \subset N(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$. Если $P^{\Omega,\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})$ является минимальной ЛЗ класса, то $\Omega = \{1,2,...,n\}, c_j^1 = \min_t\{x_{tj} : \mathbf{x}_t \in \mathrm{X}(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2})\},\ c_j^2 = \max_t\{x_{tj} : \mathbf{x}_t \in \mathrm{X}(P^{\Omega,\mathbf{c}^1,\mathbf{c}^2})\}$.

Для минимальных ЛЗ введем понятия граничного объекта и множества граничных объектов. Поскольку далее модели распознавания будут основаны на нахождении и использовании минимальных ЛЗ, в обозначениях ЛЗ мы будем опускать символ Ω.

Определение. Объект $\mathbf{x}_t$ называется граничным для минимальной ЛЗ $P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})$, если $P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}_t)=1$ и он является граничным для множества $N(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$. Множество граничных объектов обучения ЛЗ $P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})$ будем обозначать $G(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$. Множество неграничных (внутренних) объектов ЛЗ $P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x})$ будем обозначать $L(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))=\mathrm{X}(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))\setminus G(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$.

Пусть $\mathrm{P}_j=\{P_{ji}^{\mathbf{c}^{1i},\mathbf{c}^{2i}}(\mathbf{x})\}$ – множество ЛЗ класса K_j.

Вычислим оценочную матрицу $\|V\|_{lxm}$ следующим образом: $V(i,j)=\sum_{r=1}^{|\mathrm{P}_j|} g_i^r(\mathbf{x}_j)$, где

$$g_i^r(\mathbf{x}_j)=\exp(-\sum_{k=1}^{n}(x_{jk}-\mu_{ik}^r)^2/2\delta_{ik}^{r\,2}),\ \mu_{ik}^r=(\max_{x_j\in L(P_r)}x_{jk}+\min_{x_j\in L(P_r)}x_{jk})/2,$$

$\delta_{ik}^r=(\max_{x_j\in L(P_r)}x_{jk}-\min_{x_j\in L(P_r)}x_{jk})/2,$ если $L(P_r)\neq\varnothing$, и $g_i^r(\mathbf{x}_j)=0$ в противном случае.

Будем относить объект обучения $\mathbf{x}_t$ к некоторому классу K_j по следующему правилу:

1. Если $\exists P_i^{\mathbf{c}^{1i},\mathbf{c}^{2i}}\in\mathrm{P}_j: P_i^{\mathbf{c}^{1i},\mathbf{c}^{2i}}(\mathbf{x}_t)=1, \mathbf{x}_t$ – внутренний объект для $N(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$, то $\widetilde{A}_t^y(\mathbf{x}_t)=j$;
2. Если $\forall P_i^{\mathbf{c}^{1i},\mathbf{c}^{2i}}\in\mathrm{P}_j: P_i^{\mathbf{c}^{1i},\mathbf{c}^{2i}}(\mathbf{x}_t)=1, \mathbf{x}_t\in G(P^{\mathbf{c}^1,\mathbf{c}^2}(\mathbf{x}))$, то $\widetilde{A}_t^y(\mathbf{x}_t)=\arg\max_{j=1,2,\dots,l}V(j,t).$

В случае, если максимум функции $V(j,t)$ достигается на нескольких значениях j, то берем из них произвольное.

Для решения задачи оптимизации (2) предлагается следующий алгоритм приближенного пересчета множества ЛЗ $P^j, j=1..l$, а также матрицы $\|V\|_{lxm}$ при рассмотрении соседней точки произвольного шага алгоритма оптимизации.

Зафиксируем некоторое значение вектора $\mathbf{y}$ и рассмотрим соседние для него точки по координате y_i.

$$\mathbf{y}^*=(y_1^*,y_2^*,\dots,y_{l-1}^*):\begin{cases} y_j^*=y_j, & j\neq i,\\ y_j^*\neq y_j, & j=i.\end{cases}$$

Без ограничения общности будем считать, что $y_i^*<y_i$. В таком случае некоторый объект $\mathbf{x}_t\in K_i$ перейдет из одного класса в другой, т.е. $K_j^*=K_j, j\neq i,i+1$, $K_i^*=K_i\setminus\{\mathbf{x}_t\}$, $K_{i+1}^*=K_{i+1}\cup\{\mathbf{x}_t\}$,

Обозначим через $\mathrm{P}_j^*, j=1,2,\dots,l,$ и $\|V\|_{lxm}$ набор ЛЗ и оценочную матрицу, соответствующие классам $K_1^*,K_2^*,\dots,K_l^*$. Многие ЛЗ и значения матрицы не изменяются: $\mathrm{P}_j^*=\mathrm{P}_j, j\neq i,i+1$, $V^*(j,t)=V(j,t), j\neq i,i+1, t=1,2,\dots,m.$

Обозначим множество объектов обучения, для которых $P_{ji}^{\mathbf{c}^{1i},\mathbf{c}^{2i}}(\mathbf{x}_t)=1$ через $\mathrm{X}(P_{ji})$. Для вычисления «приближений» $\mathrm{P}_j^*, j=1,2,...,l,$ и $\|V^*\|_{lxm}$ в режиме скользящего контроля предлагается следующий алгоритм.

Поочередно будем перебирать все ЛЗ P_{ji} множества $\mathrm{P}_j^*, j=1,2,...,l,$ и удалять объект $\mathbf{x}_t$ из каждой ЛЗ. Если $P_{ji}(\mathbf{x}_t)=0$, тогда $P_{ji}^*(\mathbf{x}_t)=P_{ji}(\mathbf{x}_t)$. Если $P_{ji}(\mathbf{x}_t)=1$, тогда рассмотрим 3 случая:

1. $\mathbf{x}_t$ является единственным объектом на одной из границ ЛЗ.

Тогда $\mathrm{X}(P_{ji}^*)$ = $\mathrm{X}(P_{ji})\setminus\{\mathbf{x}_t\}$. Пересчитаем ЛЗ $P_{ji}^{*\mathbf{c}_j^1,\mathbf{c}_j^2}(\mathbf{x})=P_{ji}^{\widehat{\mathbf{c}}_j^1,\widehat{\mathbf{c}}_j^2}(\mathbf{x})$, где $\widehat{c}_{j\alpha}^1=\min\limits_{\mathbf{x}_q\in\mathrm{X}(P_{ji}^*)} x_{q\alpha}$, $\widehat{c}_{j\alpha}^2=\max\limits_{\mathbf{x}_q\in\mathrm{X}(P_{ji}^*)} x_{q\alpha}$, $\alpha=1,2,...,n.$

Пересчитываем $L(P_{ji}^*(\mathbf{x}))$ исходя из «новой» ЛЗ $P_{ji}^*(\mathbf{x})$, $V^*(i,t)=V(i,t)-g_j^r(\mathbf{x}_t)+g_j^{*r}(\mathbf{x}_t)$.

2. Если для некоторой ЛЗ P_{ji} имеет место $\mathbf{x}_t\in\mathrm{X}(P_{ji})$, то разбиваем $\mathrm{X}^*(P_{ji})$ на два множества по условиям $x_{q\alpha}>x_{t\alpha}$ и $x_{q\alpha}<x_{t\alpha}$, соответственно.

3. Если $\mathrm{X}(P_{ji})$ после удаления объекта $\mathbf{x}_t$ нельзя разбить на два, то рассматриваем последовательно все объекты $\mathrm{X}(P_{ji})$. Если рассматриваемый объект уже принадлежит какой-то другой ЛЗ класса, то множества ЛЗ не меняются. Если рассматриваемый объект (или несколько) не принадлежат никакому из других $\mathrm{X}(P_{ji}), i=1,2,...,|\mathrm{P}_j|,$ то пытаемся расширить все $\mathrm{X}(P_{ji})$ за счет включения этого объекта. Если объект не может быть включить ни в одно из ранее имеющихся множеств $\mathrm{X}(P_{ji})$, то строим на объекте новую ЛЗ. Таким образом, вместо множеств ЛЗ $\mathrm{P}_j, j=1,2,...,l,$ вычисляем новые множества $\mathrm{P}_j^*, j=1,2,...,l$. Соответственно, вычисляем $V^*(i,t)$.

Таким образом, для алгоритма голосования по логическим закономерностям вычисление функционала в соседней точке общего алгоритма оптимизации также осуществляется эффективно с помощью «упрощенного» приближенного пересчета множеств ЛЗ.

Результаты работы модифицированного алгоритма вычисления оценок на практических данных

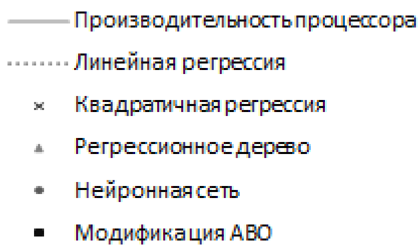
Результаты работы модифицированного алгоритма вычисления оценок сравнивались с алгоритмом линейной и квадратичной регрессии, регрессионного бинарного дерева принятия решений и двухслойной нейронной сети на реальных данных.

В качестве реальных данных рассматривалась задача “Relative CPU Performance Data” [Phillip Ein-Dor, Jacob Feldmesser, 1987] (прогнозирование производительности процессоров). Обучающая выборка состоит из 209 объектов, 8 признаков, из которых 2 признака в эксперименте не использовались (название производителя процессора и модель процессора). Таким образом, использовались 6 целочисленных признаков со следующими характеристиками:

Название признака	Минимальное значение	Максимальное значение	Среднее значение
Машинное время цикла	17	1500	203.8
Минимальная память	64	32000	2868.0
Максимальная память	64	64000	11796.1
КЭШ-память	0	256	25.2
Минимальное число каналов в процессоре	0	52	4.7
Максимальное число каналов в процессоре	0	176	18.2

Ниже на рисунках 1-2 показаны результаты работы алгоритмов на обучающей выборке. Модифицированный алгоритм вычисления оценок дает наилучший результат при числе классов l=64 (результат показан на рисунках 1 и 2). На рисунках 3-4 показаны результаты работы алгоритмов в режиме скользящего контроля leave-one-out (LOO).

Обозначения точек и графиков



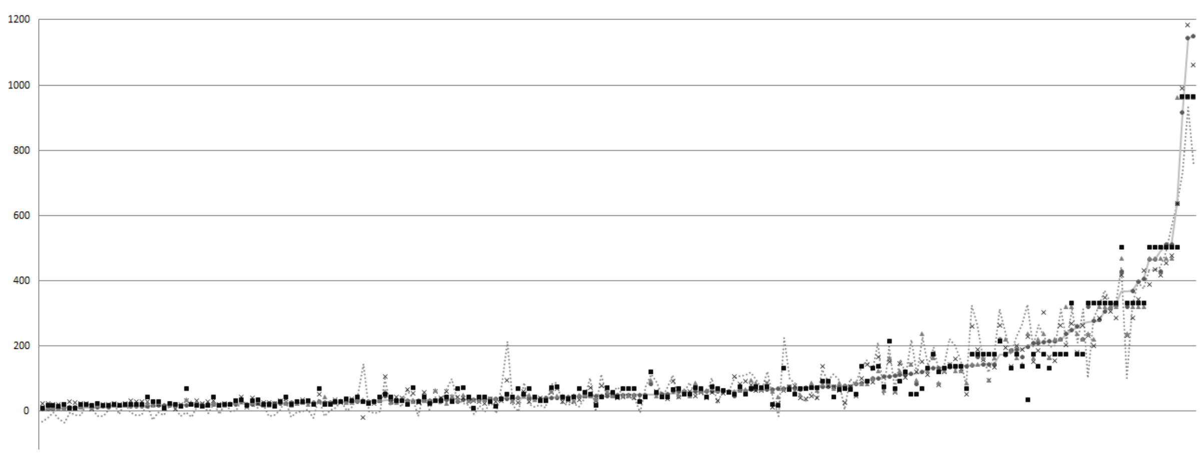


Рис. 1. Результаты прогнозирования на обучающей выборке

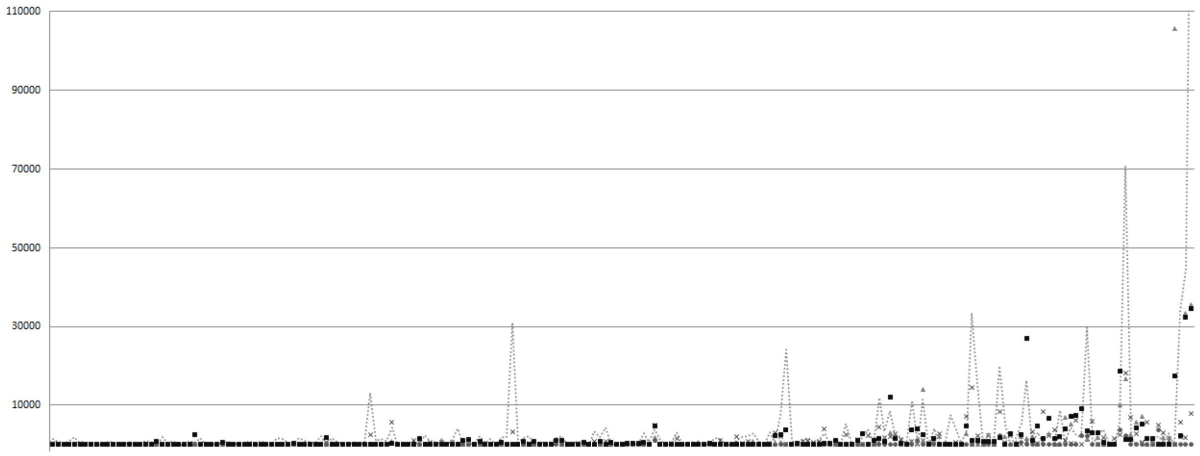


Рис. 2. Квадратичная ошибка на обучающей выборке

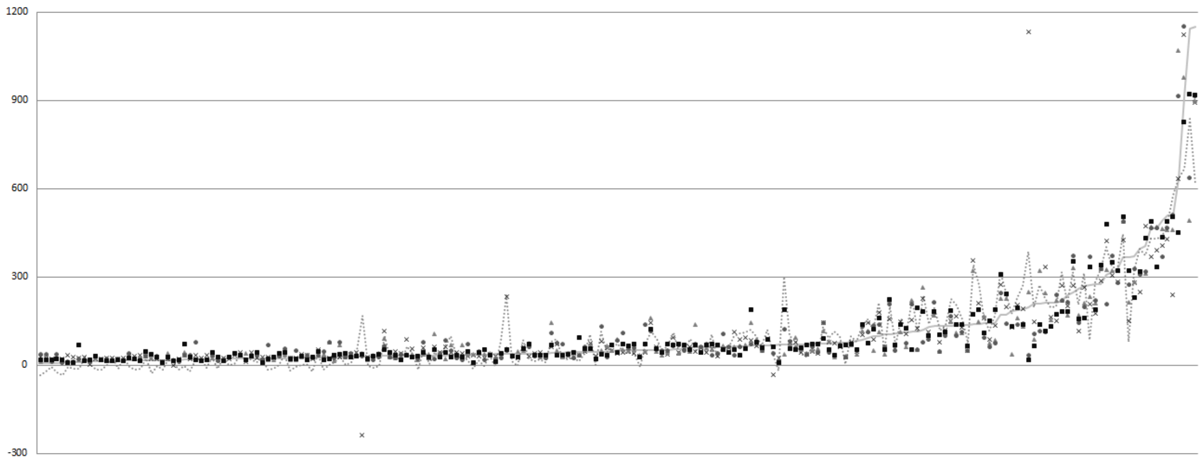


Рис. 3. Результаты прогнозирования в режиме скользящего контроля

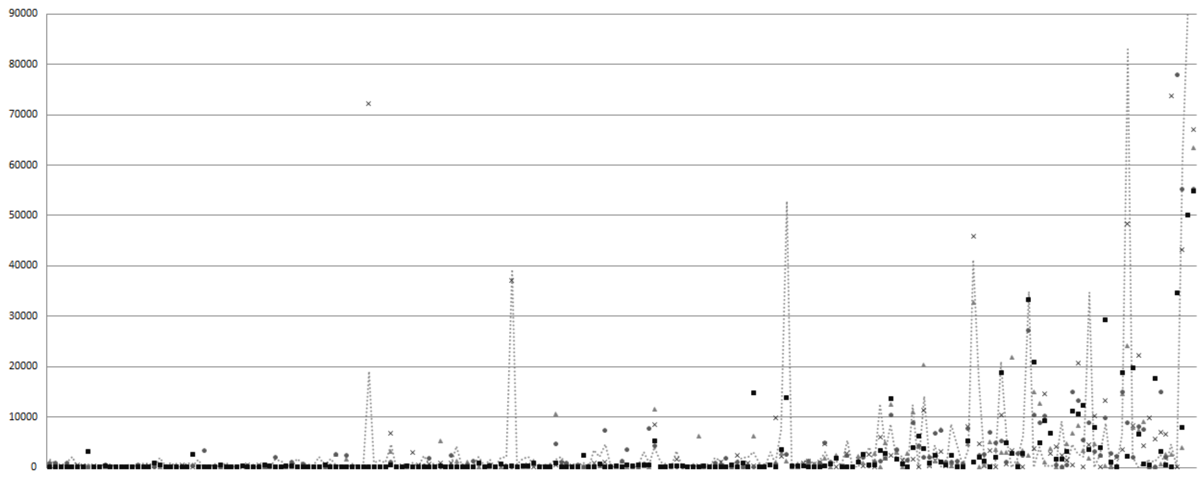


Рис. 4. Квадратичная ошибка на обучающей выборке в режиме скользящего контроля

Суммарные значения квадратичной ошибки алгоритмов по всем объектам

	Ошибка Модификации АВО	Ошибка линейной регрессии	Ошибка квадратичной регрессии	Ошибка регрессионного дерева	Ошибка нейронной сети
На обучающей выборке	295 822,56	762 326,89	220 317,75	304 987,35	20 827,23
В режиме скользящего ронтроля LOO	534 275,125	1 081 745,973	3 092 982,792	1 047 395,129	819 349,2052

При скользящем контроле модифицированный алгоритм вычисления оценок показал меньшее суммарное значение квадратичной ошибки, чем алгоритмы линейной и квадратичной регрессии, регрессионного дерева и двухслойной нейронной сети.

Заключение

Представляется важным отметить следующие детали настоящего подхода.

1. Алгоритмы восстановления регрессии рассматривались для случая числовых признаков, $x_i \in R, i = 1,2,...,n$. Легко видеть, что при построении кусочно-постоянных регрессий по обучающим выборкам признаки могут быть разнотипными: числовыми, бинарными, порядковыми. Предложенная модель восстановления регрессии основана на использовании модификации модели вычисления оценок, которая не требует наличия метрики в признаковом пространстве.
2. В начале работы отмечены некоторые общие случаи, когда классические методы восстановления регрессий плохо применимы. Здесь следует добавить случаи, когда значения зависимой величины представлены весьма неравномерно, или она является фактически k -значной величиной с большим значением k . Для задач распознавания наиболее предпочтителен случай с минимальным числом классов 2, особенно при фиксированной длине выборки. Таким образом, ожидается, что модель построения кусочно-постоянных регрессий будет полезна при решении многочисленных «плохих» задач, неудобных как для стандартных регрессионных подходов, так и для задач распознавания. В данных случаях «плохая» задача регрессии сводится к задаче распознавания с оптимальным выбором k .

В модели построения кусочно-постоянных регрессий мы полагали, что $f(\mathbf{x}) = (a_j + a_{j-1})/2$, $j = 1,2,...,l$, $a_0 = a$, $a_l = b$. Здесь конечно возможны и более точные способы непараметрического оценивания функций. Представляется перспективным их вычисление, согласованное с вычислением функций близости (8), или использование модели распознавания, основанной на голосовании по системам логических закономерностей классов [Рязанов, 2007].

3. Построение кусочно-квадратичных, кусочно-полиномиальных, и т.п. функций может осуществляться аналогично построению кусочно-линейных функций.

Acknowledgements

Настоящая работа выполнена при поддержке Программы Президиума РАН №14 , Программы №2 Отделения математических наук РАН, проектов РФФИ № 11-01-00585-а , № 12-01-90012-бел-а , № 12-01-00912-а , N 13-01-90616 -арм-а.

Библиография

[Bellman, 1957] R. Bellman, "Dynamic programming", Princeton Univ. Press (1957)

[Collobert, Bengio, 2001] R. Collobert, S. Bengio. SVMTorch: Support Vector Machines for Large-Scale Regression Problems. Journal of Machine Learning Research, 1:143-160, 2001.

[Jing-Rung Yu, Gwo-Hshiung Tzeng, Han-Lin Li 2001] Jing-Rung Yu, Gwo-Hshiung Tzeng, Han-Lin Li: General fuzzy piecewise regression analysis with automatic change-point detection. Fuzzy Sets and Systems 119(2): 247-257 (2001)

[Дрейпер, Смит, 2007] Дрейпер Н., Смит Г. Прикладной регрессионный анализ. М.: Издательский дом Вильямс, 2007.

[Журавлев, Никифоров, 1971] Журавлев Ю.И., Никифоров В.В. Алгоритмы распознавания, основанные на вычислении оценок // Кибернетика. 1971. №3. С. 1-11.

[Ковшов, Моисеев, Рязанов, 2008] Ковшов Н.В., Моисеев В.Л., Рязанов В.В. Алгоритмы поиска логических закономерностей в задачах распознавания // Журнал вычислительной математики и математической физики, Т.48, 2008, N 2, стр. 329-344.

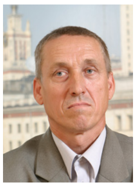
[Михалевич, 1965] Михалевич В.С. Последовательные алгоритмы оптимизации и их применение// Кибернетика. 1965, №1. С.45-46.

[Рязанов, 2007] Рязанов В.В. Логические закономерности в задачах распознавания (параметрический подход) // Журнал вычислительной математики и математической физики, Т.47, №10, 2007, с.1793-1808

[Рязанов, Тишин, Щичко, 2009] Рязанов В.В., Тишин К.В., Щичко А.С. Восстановление зависимостей по прецедентам на основе применения методов распознавания и динамического программирования// Математические методы распознавания образов: 14-я Всероссийская конференция. Владимирская обл., г.Суздаль, 21-26 сентября 2009 г.: Сборник докладов.-М.: МАКС Пресс, 2009. Стр. 168-171

[Хардле, 1993] Хардле В. Прикладная непараметрическая регрессия. М., Мир, 1993.

Информация об авторах



Vladimir Ryazanov *– Heard of Department; Institution of Russian Academy of Sciences Dorodnicyn Computing Centre of RAS, Russia, 119991 Moscow, Vavilov's street, 40;*

e-mail: rvv@ccas.ru

Major Fields of Scientific Research: Pattern recognition, Data mining, Artificial Intelligence

Anton Shchichko *– Postgraduate student; Lomonosov Moscow State University; Faculty of Computational Mathematics and Cybernetics; Department of Mathematical Forecasting Methods, Russia, 119991, Moscow, Kravchenko, 7;*

e-mail: anton.schichko@gmail.com

Major Fields of Scientific Research: Pattern recognition, Data mining, Artificial Intelligence