

# Структуры в распознавании

Н.В. Пестрякова

**Аннотация.** Изложены результаты анализа метода распознавания символов, основанного на полиномиальной регрессии, при распознавании рукопечатных цифр. Введены понятия мелко-, средне- и крупномасштабных явлений при описании данного метода распознавания. Показано наличие структур на средне- и крупномасштабном уровнях.

**Ключевые слова:** распознавание символов, полиномиальная регрессия, рукопечатная цифра, структура.

## Введение

Настоящая работа является непосредственным продолжением [5] из серии [1-5], посвященной одному современному точному методу распознавания символов, основанному на линейном регрессионном анализе [6-14]. Метод обладает высоким качеством распознавания и монотонностью оценок [2,3].

Результаты получены на базе рукопечатных цифр, которая используется как для обучения, так и для распознавания.

В [5] исследовано поведение оценки распознавания при увеличении «расстояния» между полиномиальным вектором, построенным по растрю символа, и «среднестатистическим» вектором этого символа по базе. Проведено сравнение с результатами [4], где рассматривалось «расстояние» между растром и «среднестатистическим» растром для каждого из символов.

Работа содержит новый материал, относящийся к «среднестатистическим» растрам и векторам. Выполнено обобщение этих данных и результатов работы [5].

## 1. Метод распознавания

### 1.1. Постановка задачи

Задача распознавания символов состоит в разработке алгоритма, позволяющего по данному раstrу изображения определить, какому

символу из некоторого конечного множества с  $K$  элементами он соответствует. Представлением символа является растр, состоящий из  $N=N_1 \times N_2$  серых или черно-белых пикселей. Перенумеровав все пиксели раstra, запоминаем в  $i$ -ой компоненте ( $1 \leq i \leq N$ ) вектора  $v \in \mathbf{R}^N$  состояние  $i$ -го пикселя, а именно, 0 или 1 в случае черно-белого раstra и значение на отрезке  $[0,1]$  для серого раstra. Пусть  $V=\{v\}$  – совокупность всевозможных растр. Очевидно,  $V \subseteq \mathbf{R}^N$ , причем если пиксели черно-белые, то  $V=\{0,1\}^N$  – конечное множество, элементами которого являются последовательности из нулей и единиц длины  $N$ . Если пиксели серые, то  $V=[0,1]^N$  –  $N$ -мерный единичный куб в  $\mathbf{R}^N$ .

Математическая постановка задачи распознавания состоит в следующем. Пусть для некоторого раstra  $v \in V$  можно найти  $p_k(v)$  – вероятность того, что растр изображает символ с порядковым номером  $k$ ,  $1 \leq k \leq K$ . Тогда распознанным считается символ с порядковым номером  $k_o$ , где  $p_{k_o}(v)=\max p_k(v)$ ,  $1 \leq k \leq K$ .

Для решения задачи следует вычислить вектор вероятностей  $(p_1(v), p_2(v), \dots, p_K(v))$ . Он может быть найден на основе метода наименьших квадратов.

### 1.2. Метод наименьших квадратов

Отождествим  $k$ -й символ с базисным вектором  $e_k=(0 \dots 1 \dots 0)$  (1 на  $k$ -м месте,  $1 \leq k \leq K$ ) из  $\mathbf{R}^K$ ,

причем  $\mathbf{Y} = \{\mathbf{e}_1, \dots, \mathbf{e}_K\}$ . Пусть  $p(\mathbf{v}, \mathbf{y})$  – вероятность наступления события  $(\mathbf{v}, \mathbf{y})$ ,  $\mathbf{v} \in \mathbf{V}$ ,  $\mathbf{y} \in \mathbf{Y}$  (если  $\mathbf{V}$  континуально, то плотность вероятности). Наступление события  $(\mathbf{v}, \mathbf{y})$  означает, что выпадает растр  $\mathbf{v}$ , и этот растр изображает символ  $\mathbf{y}$ . Тогда вероятность  $p_k(\mathbf{v})$  – это условная вероятность  $p_k(\mathbf{v}) = p(\mathbf{e}_k | \mathbf{v}) = p(\mathbf{v}, \mathbf{e}_k) / \sum_{s=1}^K p(\mathbf{v}, \mathbf{e}_s)$ . С другой стороны, имеет место следующий результат [6,12]:

### Теорема 1. Экстремальная задача

$$E\{\|\mathbf{d}(\mathbf{v}) - \mathbf{y}\|^2\} \xrightarrow{\mathbf{d}(\mathbf{v})} \min \quad (1)$$

достигает минимума на векторе  $\mathbf{d}(\mathbf{v}) = (p_1(\mathbf{v}), \dots, p_K(\mathbf{v}))$ , причем  $\min$  берется по всем непрерывным  $\mathbf{d}: \mathbf{V} \rightarrow \mathbf{R}^K$ .

В Теореме 1  $\|\cdot\|$  – евклидова норма в  $\mathbf{R}^K$ , и для любой функции  $f: \mathbf{V} \times \mathbf{Y} \rightarrow \mathbf{R}$  через  $E\{f(\mathbf{v}, \mathbf{y})\}$  обозначено математическое ожидание случайной величины  $f$ :

$$E\{f(\mathbf{v}, \mathbf{y})\} \underset{\mathbf{V} \times \mathbf{Y}}{\text{def}}= \int f(\mathbf{v}, \mathbf{y}) p(\mathbf{v}, \mathbf{y}) d\mathbf{v} d\mathbf{y}$$

(в случае конечного  $\mathbf{V}$  или  $\mathbf{Y}$  интегрирование по соответствующей переменной заменяется на сумму). Итак, требуемый вектор вероятностей  $(p_1(\mathbf{v}), \dots, p_K(\mathbf{v}))$  ищется как решение экстремальной задачи (1). Приближенное решение задачи (1) может быть найдено **методом полиномиальной регрессии**.

### 1.3. Метод полиномиальной регрессии

Приближенные значения компонент вектора  $(p_1(\mathbf{v}), \dots, p_K(\mathbf{v}))$  будем искать в виде многочленов от координат  $\mathbf{v} = (v_1, \dots, v_N)$ :

$$\begin{aligned} p_1(\mathbf{v}) &\equiv c_0^{(1)} + \sum_{i=1}^N c_i^{(1)} v_i + \sum_{i,j=1}^N c_{i,j}^{(1)} v_i v_j + \dots \\ p_2(\mathbf{v}) &\equiv c_0^{(2)} + \sum_{i=1}^N c_i^{(2)} v_i + \sum_{i,j=1}^N c_{i,j}^{(2)} v_i v_j + \dots \\ &\dots \\ p_K(\mathbf{v}) &\equiv c_0^{(K)} + \sum_{i=1}^N c_i^{(K)} v_i + \sum_{i,j=1}^N c_{i,j}^{(K)} v_i v_j + \dots \end{aligned} \quad (2)$$

Суммы в правых частях равенств (2) конечные и определяются выбором базисных мономов. А именно, если

$$\mathbf{x}(\mathbf{v}) = (1, v_1, \dots, v_N, \dots)^T$$

конечный вектор размерности  $L$  из выбранных и приведенных в (2) базисных мономов, упорядоченных определенным образом, то в векторном виде соотношения (2) можно записать так:

$$\mathbf{p}(\mathbf{v}) = (p_1(\mathbf{v}), \dots, p_K(\mathbf{v})) \cong \mathbf{A}^T \mathbf{x}(\mathbf{v}), \quad (2')$$

где  $\mathbf{A}$  – матрица размера  $L \times K$ , столбцами которой являются векторы  $\mathbf{a}^{(1)}, \dots, \mathbf{a}^{(K)}$ .

Каждый такой вектор составлен из коэффициентов при мономах соответствующей строки (2) (с совпадающим верхним индексом), упорядоченных так же, как в векторе  $\mathbf{x}(\mathbf{v})$ . Следовательно, приближенный поиск вектора вероятностей  $\mathbf{p}(\mathbf{v})$  сводится к нахождению матрицы  $\mathbf{A}$ . Учитывая результат Теоремы 1, следует понимать, что матрица  $\mathbf{A}$  является решением экстремальной задачи:

$$E\{\|\mathbf{A}^T \mathbf{x}(\mathbf{v}) - \mathbf{y}\|^2\} \xrightarrow{\mathbf{A}} \min, \quad (3)$$

где  $\min$  берется по всевозможным матрицам  $\mathbf{A}$  размера  $L \times K$ .

Имеет место следующий результат [4].

**Теорема 2.** *Матрица  $\mathbf{A}$  размера  $L \times K$  доставляет решение экстремальной задаче (3) тогда и только тогда, когда  $\mathbf{A}$  является решением матричного уравнения*

$$E\{\mathbf{x}(\mathbf{v}) \mathbf{x}(\mathbf{v})^T\} \mathbf{A} = E\{\mathbf{x}(\mathbf{v}) \mathbf{y}^T\}. \quad (4)$$

Итак, согласно методу полиномиальной регрессии

$$\mathbf{p}(\mathbf{v}) \cong \mathbf{A}^T \mathbf{x}(\mathbf{v}), \quad \mathbf{A} = E\{\mathbf{x}(\mathbf{v}) \mathbf{x}(\mathbf{v})^T\}^{-1} E\{\mathbf{x}(\mathbf{v}) \mathbf{y}^T\}.$$

Заметим, что если  $\mathbf{x}(\mathbf{v}) = (1, v_1, \dots, v_N)^T$ , то метод полиномиальной регрессии называется методом линейной регрессии [12].

### 1.4. Вычисление матриц

Вычисление матриц  $E\{\mathbf{x}(\mathbf{v}) \mathbf{x}(\mathbf{v})^T\}$ ,  $E\{\mathbf{x}(\mathbf{v}) \mathbf{y}^T\}$  основано на следующем результате математической статистики. Пусть имеется датчик случайных векторов, распределенных по неизвестному нам закону  $p(\mathbf{v}, \mathbf{y})$ :

$$[\mathbf{v}^{(1)}, \mathbf{y}^{(1)}], [\mathbf{v}^{(2)}, \mathbf{y}^{(2)}], \dots$$

Тогда математическое ожидание любой случайной величины  $f(\mathbf{v}, \mathbf{y})$  может быть вычислено из предельного равенства:

$$\lim_{J \rightarrow \infty} \frac{E\{f(v, y)\}}{J} = f(v^{(1)}, y^{(1)}) + f(v^{(2)}, y^{(2)}) + \dots + f(v^{(J)}, y^{(J)}).$$

Если  $J$  достаточно большое, то верно приближенное равенство:

$$\frac{E\{f(v, y)\}}{J} \approx \frac{f(v^{(1)}, y^{(1)}) + f(v^{(2)}, y^{(2)}) + \dots + f(v^{(J)}, y^{(J)})}{J}. \quad (5)$$

Набор  $[v^{(1)}, y^{(1)}], \dots, [v^{(J)}, y^{(J)}]$  практически реализуется некоторой базой данных, а вычисления по формуле (5) называются обучением. В данном случае  $f(v, y) = x_i(v)x_j(y)$  или  $f(v, y) = x_i(v)y_k$ , и по определению

$$E\{\mathbf{x}(v) \mathbf{x}(v)^T\} = [E\{x_i(v)x_j(v)\}]_{1 \leq i, j \leq L} = \\ [\sum_{s=1}^K \int_v x_i(v)x_j(v)p(v, e_s)dv]_{1 \leq i, j \leq L}$$

$$E\{\mathbf{x}(v) \mathbf{y}^T\} = [E\{x_i(v)y_k\}]_{1 \leq i \leq L, 1 \leq k \leq K} = \\ [\sum_{s=1}^K \int_v x_i(v)(e_s)_k p(v, e_s)dv]_{1 \leq i \leq L, 1 \leq k \leq K} = \\ [\sum_{s=1}^K \int_v x_i(v)p(v, e_k)dv]_{1 \leq i \leq L, 1 \leq k \leq K}$$

А согласно формуле (5) получим выражение для практического вычисления:

$$E\{\mathbf{x}(v) \mathbf{x}(v)^T\} \approx \frac{1}{J} \sum_{j=1}^J \mathbf{x}^{(j)} (\mathbf{x}^{(j)})^T, \\ E\{\mathbf{x}(v) \mathbf{y}^T\} \approx \frac{1}{J} \sum_{j=1}^J \mathbf{x}^{(j)} (\mathbf{y}^{(j)})^T, \quad (6)$$

где  $\mathbf{x}^{(j)} = \mathbf{x}(v^{(j)})$ ,  $1 \leq j \leq J$

### 1.5. Практическое нахождение матрицы А

Согласно (4) и (6) имеем следующее приближенное значение для  $A$ :

$$A \approx \left( \frac{1}{J} \sum_{j=1}^J \mathbf{x}^{(j)} (\mathbf{x}^{(j)})^T \right)^{-1} \left( \frac{1}{J} \sum_{j=1}^J \mathbf{x}^{(j)} (\mathbf{y}^{(j)})^T \right). \quad (7)$$

В [7] показано, что правую часть (7) можно вычислить по следующей рекуррентной процедуре, где  $A_0$  и  $G_0$  заданы:

$$A_j = A_{j-1} - \alpha_j G_j x^{(j)} [A_{j-1}^T x^{(j)} - y^{(j)}]^T, \quad \alpha_j = 1/J;$$

$$G_j = \frac{1}{1 - \alpha_j} [G_{j-1} - \alpha_j \frac{G_{j-1} x^{(j)} (x^{(j)})^T G_{j-1}}{1 + \alpha_j ((x^{(j)})^T G_{j-1} x^{(j)} - 1)}] \quad (8)$$

Введение вспомогательной матрицы  $G_j$  размера  $L \times L$  помогает избежать процедуры обращения матрицы в (7). Реально выбор параметров  $\alpha_j$  производится экспериментально. Вообще на практике используются следующие две упрощенные модификации процедуры (8).

#### Модификация А.

$$G_j \equiv E, \quad A_j = A_{j-1} - \frac{1}{J} x^{(j)} [A_{j-1}^T x^{(j)} - y^{(j)}]^T. \quad (9)$$

#### Модификация Б.

$$G_j \equiv D^{-1}, \quad D = \text{diag}(E\{x_1^2\}, E\{x_2^2\}, \dots, E\{x_L^2\}). \quad (10)$$

## 2. Развитие и практическая реализация метода распознавания

### 2.1. Некоторые замечания

Следует сразу отметить, что ниже рассматривается только **Модификация Б**, так как, в отличие от **Модификации А**, именно в этом случае получены приемлемые практические результаты. Поэтому не исследовалась постановка задачи с уравнениями, записанными в общем виде, и вследствие этого с существенно более медленными алгоритмами обучения и распознавания. Мы будем использовать серые распакты размера  $N=256=16 \times 16$ . Масштабирование образов до размера  $16 \times 16$  сохраняет особенности геометрии исходных символов (Рис.1).

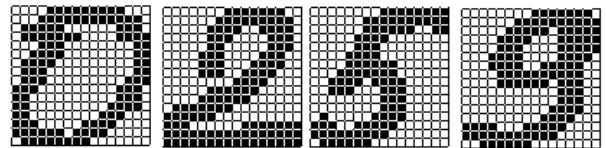


Рис. 1. Образы  $16 \times 16$  рукопечатных цифр

### 2.2. Построение вектора x

Используются два варианта вектора  $x$ : короткий и длинный. Для более длинного вектора большие размеры таблицы (матрицы) распознавания, медленнее осуществляется обучение и распознавание, но качество распознавания выше. Длинный вектор строится так:

$$\mathbf{x} = (1, \{v_i\}, \{v_i^2\}, \{(\delta v_i)_r\}, \{(\delta v_i)_r^2\}, \{(\delta v_i)_y\}, \{(\delta v_i)_y^2\}, \{(\delta v_i)_r^4\}, \{(\delta v_i)_y^4\}, \{(\delta v_i)_r(\delta v_i)_y\}, \{(\delta v_i)_r^2(\delta v_i)_y^2\}, \{(\delta v_i)_r^4(\delta v_i)_y^4\}, \{(\delta v_i)_r((\delta v_i)_r)_L\}, \{(\delta v_i)_y((\delta v_i)_y)_L\}, \{(\delta v_i)_r((\delta v_i)_y)_L\}, \{(\delta v_i)_y((\delta v_i)_r)_L\}, \{(\delta v_i)_r((\delta v_i)_D)\}, \{(\delta v_i)_y((\delta v_i)_D)\}, \{(\delta v_i)_r((\delta v_i)_y)_D\}, \{(\delta v_i)_y((\delta v_i)_r)_D\}). \quad (11)$$

Короткий вектор составлен из элементов длинного вектора в первой строке (11)

$$\mathbf{x} = (1, \{v_i\}, \{v_i^2\}, \{(\delta v_i)_r\}, \{(\delta v_i)_r^2\}, \{(\delta v_i)_y\}, \{(\delta v_i)_y^2\}). \quad (12)$$

В (11) и (12) выражения в фигурных скобках соответствуют цепочкам элементов вектора, вычисляемым по всем пикселям растра (за исключением указанных ниже случаев). Через  $(\delta v_i)_r$  и  $(\delta v_i)_y$  обозначены конечные центральные разности величин  $v_i$  по ортогональным направлениям ориентации растра – нижние индексы  $r$  и  $y$  соответственно. Если имеется нижний индекс  $L$  (left) или  $D$  (down), то это означает, что соответствующие величины относятся к пикслю слева или снизу от рассматриваемого. Компоненты вектора  $\mathbf{x}$ , не имеющие индекса  $L$  или  $D$ , вычисляются для всех пикселей растра; с индексом  $L$  – кроме левых граничных; с индексом  $D$  – кроме нижних граничных пикселей. Вне растра считаем, что  $v_i=0$  (используется при вычислении конечных разностей на границе растра). Для длинного вектора  $\mathbf{x}$  к перечисленному в (11) добавляются компоненты, являющиеся средним арифметическим значений  $v_i$  (по восьми пикселям, окружающим данный), а также квадраты этих компонент.

Отметим, что набор компонент вектора  $\mathbf{x}$  подобран в процессе численных экспериментов. А именно, среди множества рассмотренных вариантов оставлены те, которые делают заметный вклад в улучшение распознавания. Нетрудно догадаться, что способов построения и, соответственно, видов полиномиальных векторов бесконечно много.

### 2.3. Алгоритмы обучения и распознавания (модификация Б)

При обучении элементы матрицы  $\mathbf{D}$  (10) вычисляются так. Для каждого  $j$ -го элемента базы символов последовательно, начиная с первого и заканчивая  $J$ -м (последним), строится вектор

$\mathbf{x}^j$  согласно (11) или (12). Попутно рассчитываются значения компонент вспомогательного вектора  $\mathbf{m}^j$  по рекуррентной формуле:

$$\mathbf{m}_p^j = (1 - \beta_j) \mathbf{m}_p^{j-1} + \beta_j (\mathbf{x}_p^j)^2, \quad (13)$$

$$\beta_j = 1/j, \quad j=1, \dots, J, \quad p=1, \dots, L$$

По окончании этой процедуры для последнего элемента имеем согласно (10):

$$\mathbf{G}_j \equiv \mathbf{D}^{-1} = \text{diag}(1/m_1^J, 1/m_2^J, \dots, 1/m_L^J). \quad (14)$$

После того, как завершено вычисление  $\mathbf{G}_j$ , рассчитываются элементы матрицы  $\mathbf{A}_j$  (8). Выполняется еще один проход по базе и для каждого  $j$ -го элемента базы символов, начиная с первого и заканчивая  $J$ -м, строится вектор  $\mathbf{x}^j$  согласно (11) или (12). Попутно вычисляется  $\mathbf{A}_j$ :

$$\mathbf{a}_j^{pk} = \mathbf{a}_{j-1}^{pk} - \alpha_j \mathbf{x}_k^j \left( \sum_{i=1}^L \mathbf{a}_{j-1}^{ik} \mathbf{x}_i^j - \mathbf{y}_k^j \right) / m_p^J, \quad (15)$$

$$\alpha_j = 1/J$$

$$\mathbf{A}_j = [\mathbf{a}_j^{pk}], \quad j=1, \dots, J, \quad p=1, \dots, L, \quad k=1, \dots, K.$$

На этом этапе обучения закончен: матрица  $\mathbf{A}=\mathbf{A}_J$  получена.

Распознавание осуществляется следующим образом. Для серого изображения размером 16x16 пикселей строится вектор  $\mathbf{x}$  согласно (11) или (12). После этого по формуле (2'), используя  $\mathbf{A}=\mathbf{A}_J$  (15), вычисляются оценки, соответствующие каждому из возможных символов. Затем выбирается символ с максимальной оценкой. Получаемые оценки могут выходить за рамки отрезка [0,1] из-за того, что используемый метод является приближенным. Отрицательные значения искусственно обнулялись, а те, которые были больше 1, делались равными 1. Практика распознавания показала приемлемость такого довольно грубого способа коррекции оценок.

## 3. Распознавание символов обучающей базы

### 3.1. Особенности обучения и распознавания

При анализе метода распознавания аналогично [5] как обучение, так и распознавание будет проводиться на одной и той же графической базе данных, содержащей изображения и коды символов. Это делается для обеспечения

«чистоты эксперимента», поскольку результат распознавания на символьных последовательностях, «посторонних» для обучающей базы, может сильно отличаться для разных последовательностей. Данная проблема детально рассматривалась в [4].

В настоящей работе аналогично [5] при обучении и распознавании использовалась модификация длинного вектора  $\mathbf{x}$  – (11). После многократного обучения по базе в 174 778 элементов с полученной матрицей распознается 99,5% элементов (173 897 изображений распознаются верно, 881 – неверно).

Результатом распознавания образа является код символа и его целочисленная оценка, лежащая в диапазоне [1,255] (оценка 255 является наилучшей). Эта новая оценка получается следующим образом. В результате умножения оценки на 255 старый непрерывный диапазон оценок [0,1] (пп.1, 2) переходит в новый (также непрерывный) [0,255], после чего проводится дискретизация, а именно,  $[0,1] \rightarrow 1$ ,  $(1,2] \rightarrow 2, \dots, (254,255] \rightarrow 255$ .

### **3.2. Поведение средней оценки распознавания и функции распределения при отклонении от «среднестатистического» растра**

Несомненный интерес вызывает решение вопроса о том, имеются ли какие-нибудь закономерности в выставлении оценок распознавания изображений символов или оно носит абсолютно хаотический характер. Аналогично, представляется важным понять характер распределения изображений символов.

На Рис.2а и Рис.3а (все рисунки приведены в Приложении) представлены диаграммы зависимости средней оценки распознавания символа (9 и 1 соответственно) от величины отклонения между его растрами и «среднестатистическим» растром этого символа. Для цифр 0, 2, 3, 4, 5, 6, 7, 8 диаграммы аналогичны приведенной на Рис.2а.

Для «среднестатистического» растра конкретного символа значение в пикселе, имеющем номер  $i$ , равно среднему арифметическому значений  $i$ -х пикселей по всем растрам этого символа. Расстояние между растрами  $\mathbf{v}=(v_1, \dots, v_N)$  и  $\mathbf{u}=(u_1, \dots, u_N)$  определяется так: вычисляется модуль разности значений в  $i$ -х

пикселях, затем производится суммирование по всем  $N$  пикселям:

$$\|\mathbf{v}-\mathbf{u}\| = \sum_{i=1}^N |v_i - u_i|. \quad (16)$$

Диапазон отклонений между растрами распознанных верно изображений символа и его «среднестатистическим» растром находится от минимального  $v_{true\_min}$  до максимального  $v_{true\_max}$ . В Табл. 1 приведены эти значения.

Отрезок  $[v_{true\_min}, v_{true\_max}]$  (оси абсцисс на Рис.2а, Рис.3а) был поделен на 20 равных по длине частей – отрезок и 19 полуинтервалов:  $[v_{true\_min}, v_{true\_min} + dv]$ ,  $(v_{true\_min} + dv, v_{true\_min} + 2dv]$ , …,  $(v_{true\_min} + 19dv, v_{true\_min} + 20dv]$ , где  $dv = (v_{true\_max} - v_{true\_min})/20$ . Затем для совокупности изображений, попадающих в каждый такой участок, вычислялась средняя оценка распознавания (оси ординат на Рис. 2а, Рис. 3а). Она не убывает монотонно по мере «удаления» растров от «среднестатистического», а для «1» принимает значение 255 на наибольшем удалении от «среднестатистического» образа. Диаграмма 2а и аналогичные для цифр 0, 2, 3, 4, 5, 6, 7, 8 имеют тенденцию к убыванию, но велик уровень шумов.

На Рис. 2б, Рис. 3б приведены диаграммы – «дискретный» аналог функции распределения для распознанных верно изображений символов 9, 1 (для остальных цифр аналогично). Ось абсцисс такая же, как указано выше, а по оси ординат отложено количество правильно распознанных изображений, попавших в каждую двадцатую часть отрезка  $[v_{true\_min}, v_{true\_max}]$ .

Для неправильно распознанных образов символа диапазон отклонений между его растрами и «среднестатистическим» растром этого символа по рассматриваемой базе находится от минимального  $v_{false\_min}$  до максимального  $v_{false\_max}$ . В Табл. 1 приведены значения этих величин для символов 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, а также средняя оценка неправильного распознавания  $P_{false}$ . Они приблизительно в два раза меньше, чем оценки правильного распознавания. Если сравнить  $v_{true\_min}$  некоторого символа и  $v_{false\_min}$  любого другого (или того же) символа, то  $v_{true\_min} < v_{false\_min}$ .

Табл. 1

символ	v_true_min	v_true_max	v_true_max/v_true_min	v_false_min	v_false_max	$\lceil v_{\min}$	$\lceil v_{\max}$	P <sub>false</sub>
0	35,41	113,59	3,21	56,34	101,09	50,91	171,30	106,61
1	42,58	173,80	4,08	52,22	131,38	48,00	142,30	128,17
2	38,33	105,57	2,75	61,62	109,43	49,74	168,44	120,19
3	39,92	103,10	2,58	55,63	95,53	47,68	171,61	118,84
4	50,76	106,34	2,09	56,02	123,47	55,87	172,25	131,78
5	36,28	130,66	3,60	52,68	98,26	45,89	178,17	126,65
6	44,60	115,07	2,58	55,17	103,11	55,32	165,89	105,26
7	40,56	101,70	2,51	53,45	93,89	49,42	169,91	114,43
8	50,15	119,80	2,39	57,23	115,19	53,73	157,66	121,27
9	47,36	120,58	2,55	54,28	117,37	54,81	164,77	127,45

Табл. 2

символ	x_true_min	x_true_max	x_true_max/x_true_min	x_false_min	x_false_max	$\lceil x_{\min}$	$\lceil x_{\max}$
0	2004	5290	3,21	3002	5119	2186	7754
1	2416	7917	4,08	3046	6437	2698	6915
2	2237	5265	2,75	3491	5523	2860	7710
3	2276	4954	2,58	3026	4936	2697	7723
4	2798	5158	2,09	3135	5619	3148	7729
5	2104	6300	3,60	3142	4909	2558	8057
6	2416	5161	2,58	3375	4966	3004	7437
7	2324	5276	2,51	2913	5621	2788	7645
8	2679	5505	2,39	3335	5188	2901	6977
9	2559	5482	2,55	3076	5313	2964	7458

Для произвольного символа G из рассматриваемых цифр 0, ..., 9 диапазон отклонений между растрями изображений всех символов, кроме G, и «среднестатистическим» растром цифры G по рассматриваемой базе находится от минимального  $\lceil v_{\min}$  до максимального  $\lceil v_{\max}$ . В Табл. 2 приведены значения этих величин для каждого из возможных символов G: 0, ..., 9. Отметим, что при сравнении  $v_{\text{false\_max}}$  некоторого символа и  $\lceil v_{\max}$  произвольного символа, имеем:  $v_{\text{false\_max}} < \lceil v_{\max}$ .

Кроме того, любая из трех «минимальных» величин ( $v_{\text{true\_min}}$ ,  $v_{\text{false\_min}}$ ,  $\lceil v_{\min}$ ) некоторого символа меньше, чем любая «максимальная» величина ( $v_{\text{true\_max}}$ ,  $v_{\text{false\_max}}$ ,  $\lceil v_{\max}$ ) произвольного символа.

### 3.3. Поведение средней оценки распознавания и функции распределения при отклонении от «среднестатистического» вектора

На Рис.2в и Рис.3в даны диаграммы зависимости средней оценки распознавания символа (9 и 1 соответственно) от величины отклонения между полиномиальным вектором x, построен-

ным по его раству, и «среднестатистическим» полиномиальным вектором этого символа. Для остальных цифр диаграммы аналогичны Рис.2в.

Для «среднестатистического» полиномиального вектора конкретного символа значение в каждой компоненте вектора, имеющей номер i, равно среднему арифметическому значений i-х компонент по всем растрам рассматриваемого символа. Расстояние между векторами  $v=(v_1, \dots, v_L)$  и  $u=(u_1, \dots, u_L)$  определяется так: вычисляется модуль разности значений в i-х компонентах и проводится суммирование по всем L компонентам:

$$\|v-u\| = \sum_{i=1}^L |v_i - u_i| \quad (17)$$

Диапазон отклонений между полиномиальным вектором распознанного верно изображения символа и «среднестатистическим» вектором этого символа по рассматриваемой базе лежит от минимального  $x_{\text{true\_min}}$  до максимального  $x_{\text{true\_max}}$ . В Табл. 2 приведены значения этих величин.

Отрезок  $[x_{\text{true\_min}}, x_{\text{true\_max}}]$  (оси абсцисс на Рис.2в, Рис.3в) был поделен на 20 рав-

ных по длине частей – отрезок и 19 полуинтервалов:  $[x_{true\_min}, x_{true\_min} + xdv]$ ,  $(x_{true\_min} + xdv, x_{true\_min} + 2xdv]$ , ...,  $(x_{true\_min} + 19xdv, x_{true\_min} + 20xdv]$ , где  $xdv = (x_{true\_max} - x_{true\_min})/20$ . Затем для совокупности изображений, имеющих полиномиальные векторы, попадающие в каждый такой участок, вычислялась средняя оценка распознавания (оси ординат на Рис. 2в, Рис.3в). На этих рисунках видно, что средняя оценка распознавания для каждого из рассматриваемых символов на соответствующем этому символу отрезке  $[x_{true\_min}, x_{true\_max}]$  убывает монотонно (с некоторыми шумовыми погрешностями) по мере «удаления» от «среднестатистического» вектора, а для «1» сначала монотонно убывает, а затем монотонно увеличивается и принимает максимальное значение 255 на предпоследнем интервале удаления от «среднестатистического» вектора (также с некоторыми погрешностями). Уровень шумов в этих зависимостях существенно ниже, а степень монотонности выше, чем в аналогичных зависимостях для средней оценки распознавания при отклонении от «среднестатистического» растра.

На Рис.2г, Рис.3г приведены диаграммы – «дискретный» аналог функции распределения для распознанных верно изображений символов 0, 1 (для 2, ..., 9 аналогично). Ось абсцисс здесь также, как указано в предыдущем абзаце для Рис.2в, Рис.3в, а по оси ординат отложено количество правильно распознанных изображений, попавших в каждую двадцатую часть отрезка  $[x_{true\_min}, x_{true\_max}]$ .

Для «1» при увеличении отклонения от 0 сначала наблюдалось уменьшение оценки, а затем ее рост до максимальной оценки 255 на наиболее удаленных участках. Возникла гипотеза о том, что база единиц составлена из двух подбаз.

Дальнейшие действия были проделаны для «среднестатистических» векторов, поскольку в использующих их зависимостях уровень шумов существенно ниже, чем для «среднестатистических» растров. Найдя первоначальное «среднестатистическое» значение полиномиального вектора  $x_0$ , мы отделили ту часть изображений, полиномиальные векторы которых удалены от соответствующего  $x_{true\_min}$  более чем на  $2/3$  величины  $x_{true\_max}-x_{true\_min}$ , и построили

для них «среднестатистический» вектор  $x_1$ . Оказалось, что для изображений, векторы которых к  $x_1$  ближе, чем к  $x_0$ , при удалении от  $x_1$  оценка монотонно падает (Рис.3ж). Всего оказалось 714 таких изображений. Для оставшихся 32388 изображений построили новый среднестатистический вектор  $x_2$ , при отклонении от которого имела место аналогичная закономерность (Рис.3д). Для каждой из выделенных подбаз построили функции распределения (соответственно Рис.3з, Рис.3е), которые оказались схожи с функциями распределения других символов. Дополнительные итерации, несомненно, улучшили бы степень разделения подбаз.

Проанализировано, как соотносится монотонный характер убывания средней оценки распознавания при удалении от «среднестатистического» вектора с функциями распределения верно распознанных изображений для различных оценок распознавания. Изучение указанных закономерностей проведено на примере символа «8». Рассмотрены следующие диапазоны оценок распознавания: [255, 250), [250, 240), [240, 230), [230, 220), [220, 210), [210, 200), [200, 190), [190, 180), [180, 170), [170, 160), [160, 150), [150, 140), [140, 130), [130, 120). Проводить исследования для более низких оценок нецелесообразно, поскольку количество оставшихся изображений, не попавших в описанные полуинтервалы, невелико и сопоставимо с количеством неправильно распознанных растров. Для изображений, получивших оценку распознавания внутри каждого из этих полуинтервалов, построены соответствующие функции распределения. На Рис.4 (а,б,в,г,д) приведены графики – «дискретный» аналог описанных функций распределения. Здесь ось абсцисс такая же, как указано ранее для Рис.2в, Рис.3в, Рис.2г, Рис.3г, а по оси ординат отложено количество правильно распознанных изображений, получивших оценку из соответствующего диапазона и попавших в каждую вышеописанную двадцатую часть отрезка  $[x_{true\_min}, x_{true\_max}]$ .

На Рис.4а приведен такой же график для всего «совокупного» спектра оценок распознавания, полученных изображениями цифры 8 (соответствующий диаграмме, аналогичной Рис. 2г, Рис.3г), обозначенный в «легенде» диа-

пазоном «0-255». Здесь же имеются графики функций распределения для диапазонов [255, 250] и [250, 240], указанные в «легенде» как «250-255» «240-250» соответственно. Использование диапазона [255, 250], который вдвое меньше каждого из указанных на рис. 4б, 4в, 4г, 4д объясняется тем, что в этот полуинтервал попадает большое количество символов, как это нетрудно увидеть на рис.4а при сравнении графиков «250-255» и «240-250».

В серии Рис.4 (а,б,в,г) для удобства в проведении сопоставления каждый предыдущий и последующий рисунки содержат по одному общему графику. Так для Рис.4а, Рис.4б общим является «240-250», Рис.4б, Рис.4в объединяет «210-220», а Рис.4в, Рис.4г роднит «180-190», а Рис.4г, Рис.4д - «150-160».

Итак, на каждом из двадцати частей отрезка  $[x_{true\_min}, x_{true\_max}]$  средняя оценка получается суммированием оценок 1, 2, 3, ..., 254, 255 с весами, определяемыми средней величиной (по этому отрезку) функции вероятности соответствующей оценки. Отсюда вытекает, что полученный результат, а именно, монотонное убывание средней оценки распознавания, далеко не очевиден. Он соответствует наличию структуры в ряду случайных событий.

Диапазон отклонений между полиномиальным вектором неправильно распознанного изображения символа и «среднестатистическим» вектором этого символа по рассматриваемой базе находится от минимального  $x_{false\_min}$  до максимального  $x_{false\_max}$ . В Табл. 2 приведены значения этих величин для каждого из символов 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Для произвольного символа G из рассматриваемых цифр 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 диапазон отклонений между полиномиальными векторами изображений всех символов, кроме G, и «среднестатистическим» вектором цифры G по рассматриваемой базе находится от минимального  $\lceil x_{min} \rceil$  до максимального  $\lceil x_{max} \rceil$ . В Табл. 2 приведены значения этих величин для каждого из возможных символов G: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Отметим, что для каждой из рассматриваемых цифр  $x_{true\_min} < \lceil x_{min} \rceil < x_{false\_min}$ ; кроме того,  $x_{true\_max} < \lceil x_{max} \rceil$ ; а также  $x_{false\_max} < \lceil x_{max} \rceil$ . Следовательно, для каждого символа G можно выделить полу-

интервал  $[x_{true\_min}, \lceil x_{min} \rceil]$ , в котором находятся только правильно распознанные изображения этого символа; внутри полуинтервала  $[\lceil x_{min} \rceil, x_{false\_min}]$  имеются правильно распознанные изображения символа G и изображения символов, отличных от G; в полуинтервале  $(\text{MAX}(x_{true\_max}, x_{false\_max}), \lceil x_{max} \rceil]$  лежат только изображения символов, отличных от G; внутри отрезка  $[x_{false\_min}, \text{MIN}(x_{true\_max}, x_{false\_max})]$  содержатся любые изображения (и правильно, и неправильно распознанные изображения символа G, а также отличные от G символы); в полуинтервале  $(\text{MIN}(x_{true\_max}, x_{false\_max}), \text{MAX}(x_{true\_max}, x_{false\_max})]$  - отличные от G символы, а также при  $x_{true\_max} > x_{false\_max}$  (для цифр 0, 1, 3, 5, 6, 8, 9)- правильно, а при  $x_{true\_max} < x_{false\_max}$  (для 2, 4, 7) - неправильно распознанные изображения символа G (соответственно Рис.5а, Рис. 5б).

Итак, при удалении от «среднестатистического» полиномиального вектора любого рассматриваемого символа выявлено структурированное расположение правильно и неправильно распознанных изображений рассматриваемого символа, а также отличных от него символов.

Отметим, что соотношения, приведенные в двух предыдущих абзацах, не выполняются для приведенных в Табл. 1 аналогичных величин в терминах растротов  $(v_{true\_min}, \lceil v_{min} \rceil, v_{false\_min}, v_{true\_max}, v_{false\_max}, \lceil v_{max} \rceil)$ . Следовательно, выявленное для полиномиальных векторов структурированное расположение символов при удалении от «среднестатистического» вектора не имеет места в терминах растротов.

## Заключение

Из представленного в п.п. 3.1.-3.3. материала следует, что при описании метода распознавания необходимо различать мелкомасштабные, среднемасштабные и крупномасштабные явления.

К мелкомасштабным относятся те, при описании которых вообще не используется механизм осреднения. Сюда нужно причислить распознавание и выставление оценок отдельным изображениям символов.

К среднемасштабным следует отнести те, при описании которых принципиальным является наличие (или построение) «среднестатистических» растротов и векторов различных сим-

волов, но не используется механизм осреднения оценок (или он является несущественным, служит каким-то второстепенным целям, и от него легко отказаться). Сюда нужно причислить: 1) получение «среднестатистических» растротов и векторов различных символов, 2) построение функций распределения (не обязательно осредненных по частичным диапазонам отклонения – это делалось исключительно для удобства рисования картинок) при отклонении от «среднестатистических» растротов / векторов, причем как для всей совокупности оценок, так и для отдельных оценок или их частичных диапазонов, 3) наличие структуры при отклонении от «среднестатистического» вектора символа правильно, неправильно распознанных изображений этого символа, а также «чужих» символов, 4) решение ориентироваться на полиномиальные векторы, а не на раstry, поскольку использование последних сопряжено с присутствием существенных шумовых помех. Помимо указанных четырех пунктов могут быть их комбинации и модификации. Следует только отметить, что среднемасштабные явления подразделяются на те, в которых используются «среднестатистические» раstry / векторы различных символов – они «ближе» к мелкомасштабным, и на ориентированные только на «среднестатистический» растр / вектор одного символа – эти ближе к крупномасштабным. Такое деление на две подгруппы сделать нетрудно. Уже на этом среднемасштабном уровне над «хаосом» мелкомасштабных явлений выявлено наличие структуры – в расположении правильно, неправильно распознанных, а также «чужих» символов при удалении от «среднестатистического» вектора.

К крупномасштабным явлениям относятся явления, при описании которых принципиальным является ориентация на «среднестатистический» растр или вектор одного определенного символа (о наличии остальных символов следует забыть), кроме того, используется механизм осреднения оценок. Удивительным оказывается то, что над «хаосом» мелкомасштабных и среднемасштабных явлений обнаруживается некий «порядок»: несмотря на то, что на всем диапазоне отклонений от «среднестатистического» вектора имеются символы, получившие различные оценки распознавания,

количество которых определяется уже описанными функциями распределения, результатом их «коллективного» действия оказывается наличие структуры, а именно, монотонное уменьшение средней оценки распознавания при удалении от «среднестатистического» вектора.

Среднемасштабное описание распознавания можно получить, если абстрагироваться от взаимного расположения «среднестатистических» векторов, не отрицая в то же время их наличия. Итак, есть «среднестатистический» вектор некоторого символа. К нему «привязаны» (иначе, от него отсчитываются) зоны нахождения правильно распознанных «своих» символов, неправильно распознанных «своих» символов, чужих символов (это среднемасштабная структура). Следующий шаг – осреднение по оценкам (для интервалов по признаку удаленности от «среднестатистического» вектора) – дает крупномасштабное описание распознавания. А именно, при удалении от «среднестатистического» вектора средняя оценка распознавания монотонно падает (крупномасштабная структура).

## Литература

- Гавриков М.Б., Пестрякова Н. В. "Метод полиномиальной регрессии в задачах распознавания печатных и рукопечатных символов", //Препринт ИПМатем. РАН, М., 2004, №22, 12 с.
- Гавриков М.Б., Пестрякова Н. В., Славин О.А., Фарсобина В.В.. "Развитие метода полиномиальной регрессии и практическое применение в задаче распознавания", //Препринт ИПМатем. РАН, М., 2006, №25, с 21.
- Гавриков М.Б., Мисюров А.В., Пестрякова Н.В., Славин О.А. Развитие метода полиномиальной регрессии и практическое применение в задаче распознавания символов. Автоматика и Телемеханика. 2006, №3, С. 119-134.
- Гавриков М.Б., Пестрякова Н. В., Усков А.В., Фарсобина В.В. "О некоторых свойствах метода распознавания символов, основанного на полиномиальной регрессии", //Препринт ИПМатем. РАН, М., 2007, №69, 20 с.
- Гавриков М.Б., Пестрякова Н. В., Усков А.В., Фарсобина В.В. "О некоторых свойствах оценки метода распознавания символов, основанного на полиномиальной регрессии", //Препринт ИПМатем. РАН, М., 2008, №7, 28 с.
- Sebestyen G.S. Decision Making Processes in Pattern Recognition, MacMillan, New York, 1962.
- Nilson N. J. Learning Machines, McGraw-Hill, New York, 1965.
- Schürmann J. Polynomklassifikatoren, Oldenbourg, München, 1977.

9. Schürmann J. Pattern Classification, John Wiley&Sons, Inc., 1996.
  10. Albert A.E. and Gardner L.A. Stochastic Approximation and Nonlinear Regression // Research Monograph 42. MIT Press, Cambridge, MA, 1966.
  11. Becker D. and Schürmann J. Zur verstrkten Berucksichtigung schlecht erkennbarer Zeichen in der Lernstichprobe // Wissenschaftliche Berichte AEG-Telefunken 45, 1972, pp. 97 – 105.
  12. Pao Y.-H. The Functional Link Net: Basis for an Integrated Neural-Net Computing Environment // in Yoh-Han Pao (ed.)
- Adaptive Pattern Recognition and Neural Networks, Addison-Wesley, Reading, MA, 1989, pp. 197-222.
13. Franke J. On the Functional Classifier, in Association Francaise pour la Cybernetique Economique et Technique (AFCET), Paris // Proceedings of the First International Conference on Document Analysis and Recognition, St. Malo, 1991, pp.481-489.
  14. Дж.Себер. Линейный регрессионный анализ. М.: "Мир", 1980.
  15. Линник Ю.В. Метод наименьших квадратов и основы математико - статистической теории обработки наблюдений. М.: "Физматлит", 1958.

**Пестрякова Надежда Владимировна.** Старший научный сотрудник Института системного анализа РАН. Окончила Московский физико-технический институт в 1983году. Кандидат физико-математических наук. Имеет 21 публикацию. Область научных интересов – математическая физика, численные методы, искусственный интеллект. Эл. адрес: nadya\_p@cs.isa.ru.

## Приложение

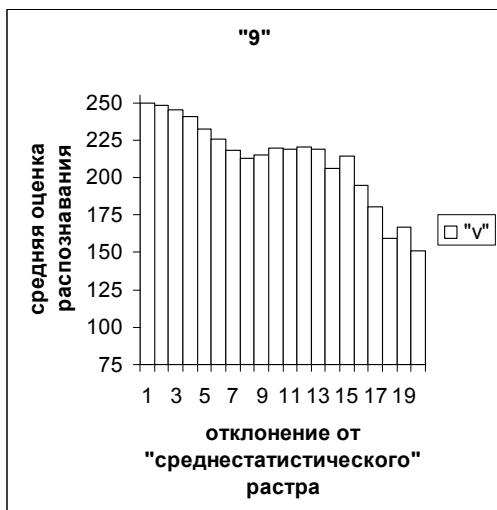


Рис. 2а

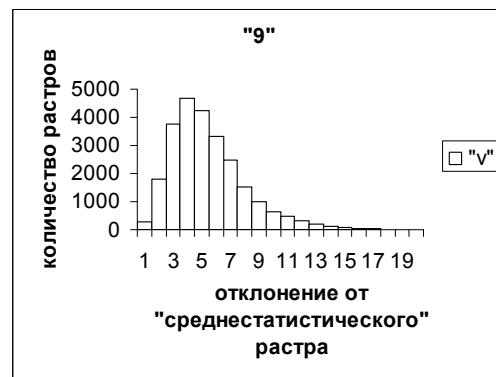


Рис. 2б

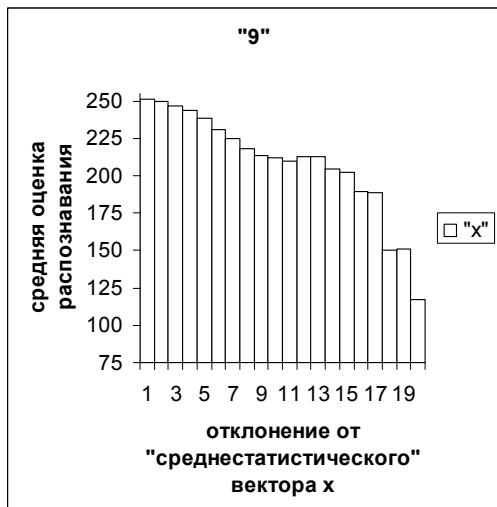


Рис. 2в

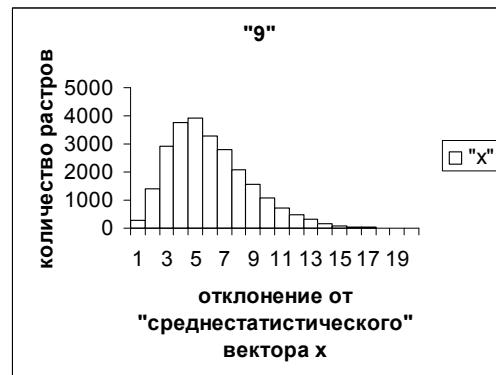


Рис. 2г

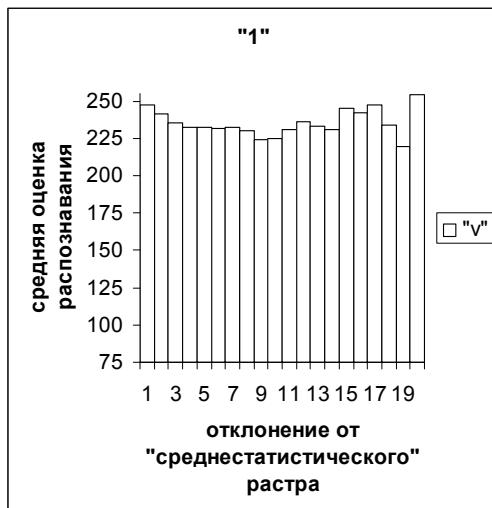


Рис. 3а

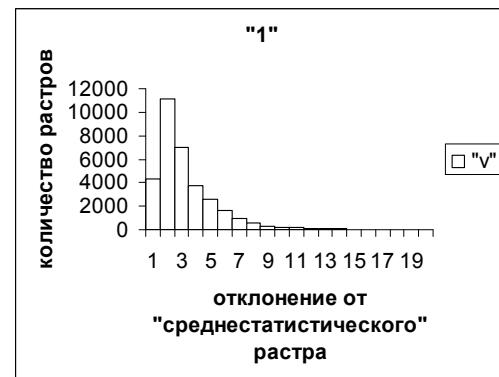


Рис. 3б

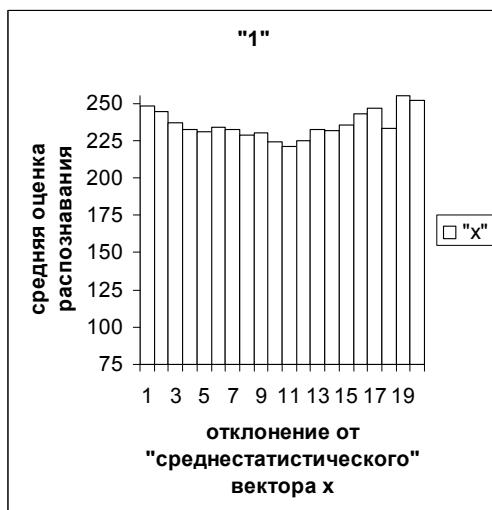


Рис. 3в

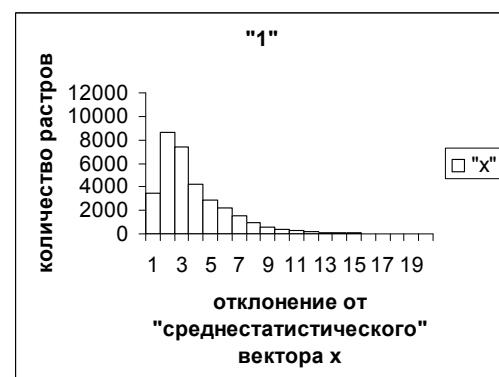


Рис. 3г

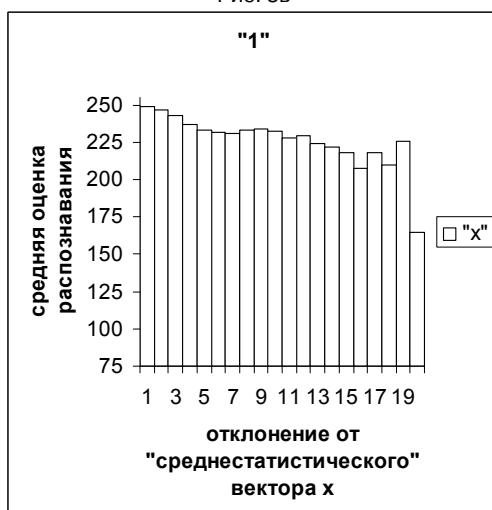


Рис. 3д

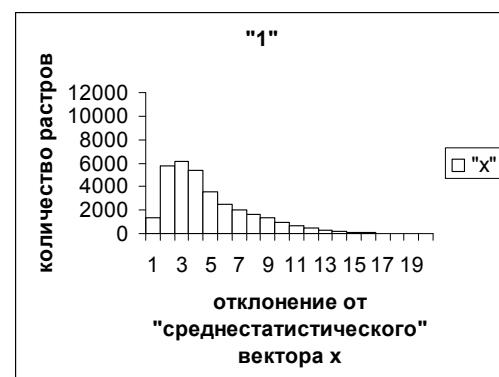


Рис. 3е

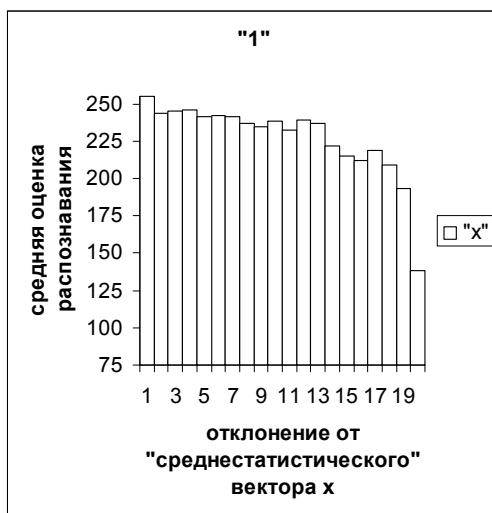


Рис. 3ж

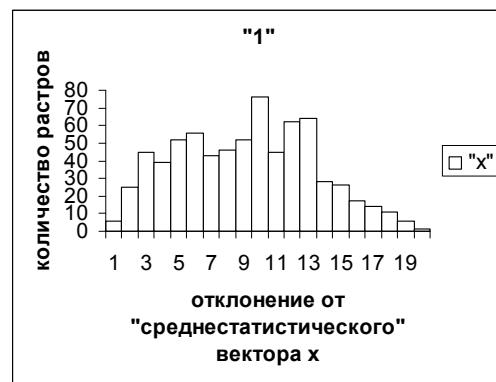


Рис. 3з

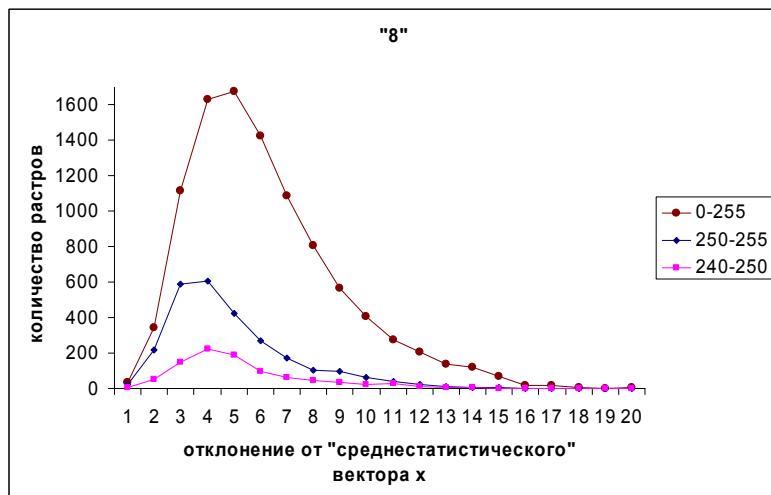


Рис. 4а

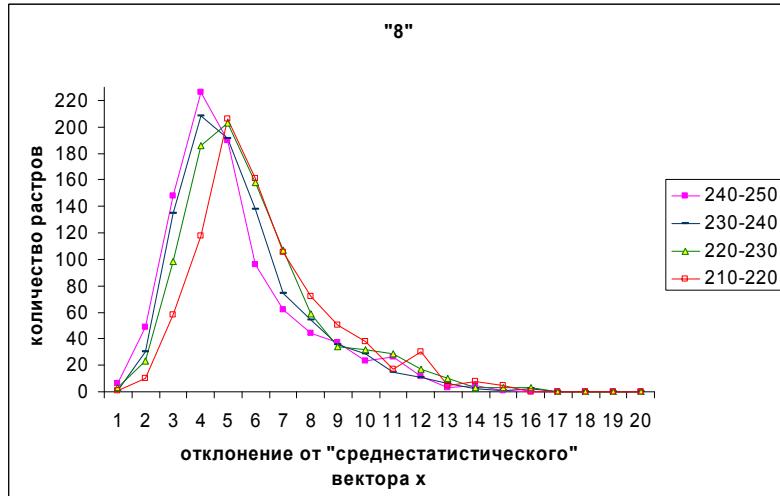


Рис. 4б

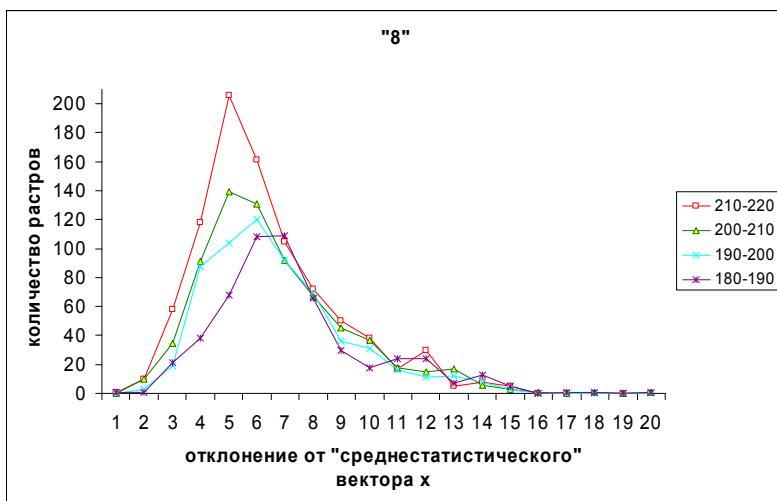


Рис. 4в

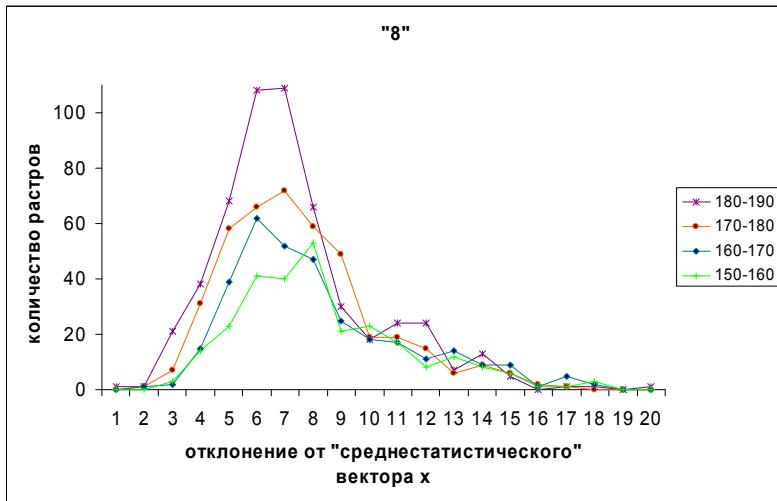


Рис. 4г

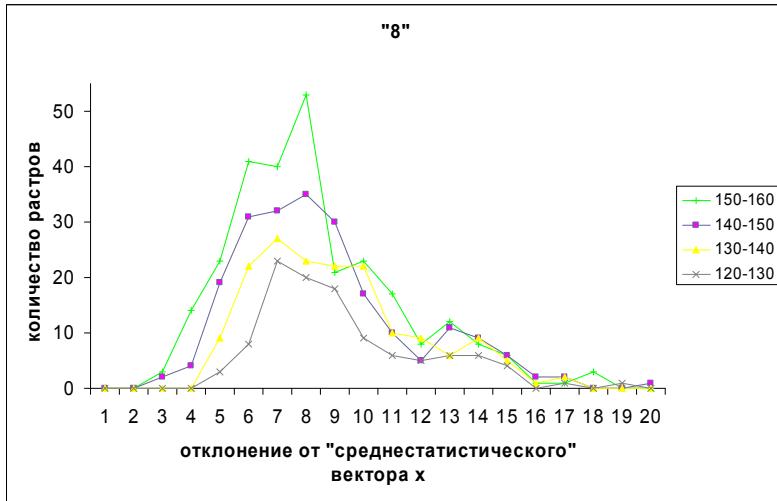


Рис. 4д

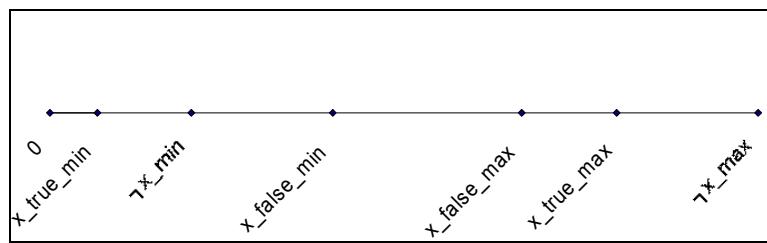


Рис. 5а

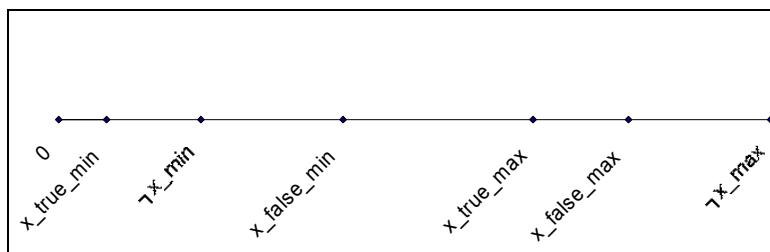


Рис. 5б