

Министерство образования и науки Российской Федерации

Федеральное государственное образовательное учреждение  
Высшего профессионального образования

Тульский государственный университет

Кафедра автоматике и телемеханики  
(наименование выпускающей кафедры)

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**  
***Выпускная квалификационная работа магистра***

направление 230100  
(код)

Информатика и вычислительная техника  
(наименование)

программа подготовки 23010019  
(код)

Компьютерный анализ и интерпретация данных  
(наименование)

Исследование модели логистической регрессии в задаче  
распознавания образов для нестационарной генеральной  
совокупности  
(тема)

Студент группы 240661/19 \_\_\_\_\_ Турков П.А.  
(подпись, дата) (фамилия, инициалы)

Научный руководитель \_\_\_\_\_ Красоткина О.В.  
(подпись, дата) (фамилия, инициалы)

Заведующий кафедрой \_\_\_\_\_ Фомичев А.А.  
(подпись, дата) (фамилия, инициалы)

Руководитель  
магистерской программы \_\_\_\_\_ Копылов А.В.  
(подпись, дата) (фамилия, инициалы)

Тула 2012

### **Аннотация**

Существуют задачи обучения распознаванию образов, в которых влияние некоторого скрытого фактора приводит к изменению свойств генеральной совокупности. Данная работа предлагает математическое и алгоритмическое описание для задач распознавания такого вида.

Представленное вероятностное обоснование основано на байесовском подходе к методу логистической регрессии для нахождения параметров решающего правила. Полученная процедура распознавания образов построена на общем принципе динамического программирования и обладает линейной вычислительной сложностью в противоположность полиномиальной для общего случая процедуры распознавания.

Пояснительная записка содержит 53 страницы, 4 рисунка, 4 таблицы.

## Содержание

Введение .....	6
1. Проблема обучения распознаванию образов в условиях нестационарной генеральной совокупности .....	8
2. Постановка задачи обучения распознаванию образов в нестационарной генеральной совокупности .....	9
2.1. Классическая задача обучения распознаванию образов .....	9
2.2. Задача обучения распознаванию образов в нестационарной генеральной совокупности .....	10
2.3. Вероятностное описание нестационарности параметров решающего правила .....	14
3. Существующие методы обучения распознаванию образов в условиях нестационарной генеральной совокупности .....	16
4. Обобщение метода опорных векторов для нестационарной генеральной совокупности .....	20
4.1. Метод опорных векторов в классической задаче обучения распознаванию образов .....	20
4.2. Вероятностное обоснование метода опорных векторов.....	22
4.3. Распространение метода опорных векторов на случай нестационарной генеральной совокупности .....	25
4.4. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода опорных векторов.....	27
4.5. Численная реализация процедуры динамического программирования для оценивания параметров оптимальной разделяющей гиперплоскости ..	28
5. Обобщение модели логистической регрессии на случай нестационарной генеральной совокупности .....	36
5.1. Модель логистической регрессии в задаче обучения распознаванию образов.....	36

5.2. Вероятностное обоснование модели логистической регрессии .....	37
5.3. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода логистической регрессии.....	39
5.4. Численная реализация процедуры динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью модели логистической регрессии .....	41
6. Экспериментальное исследование .....	47
6.1. Экспериментальное исследование на модельных данных .....	47
6.2. Экспериментальное исследование на реальных данных.....	48
Заключение .....	52
Список использованных источников .....	53

## Введение

В последние годы распознавание образов находит все большее применение в повседневной жизни. Распознавание речи и рукописного текста значительно упрощает взаимодействие человека с компьютером, распознавание печатного текста используется для перевода документов в электронную форму.

В классической постановке задачи распознавания универсальное множество, называемое генеральной совокупностью, разбивается на части-образы, также называемые классами. Образ какого-либо объекта задается набором его частных проявлений. Методика отнесения элемента к какому-либо образу называется решающим правилом. В данной работе мы будем рассматривать задачу обучения с учителем, т.е. для построения решающего правила будет использоваться некоторое множество объектов, на которых известна их скрытая характеристика –образ (класс), к которому данный объект относится. Наиболее известной и изученной является следующая ситуация: распознавание производится на множестве выбранных из генеральной совокупности объектов, свойства которых не изменяются со временем. Очевидно, что в реальных условиях это не так, но часто изменениями характеристик объектов можно пренебречь, поэтому данный подход вполне применим в большей части практических случаев. Однако, при прогнозировании сложных явлений, например, социально-экономических или биологических процессов, предположение о стационарности ведет к недопустимо большой ошибке при классификации.

Целью данной работы является исследование модели логистической регрессии с целью последующего создания алгоритма обучения распознаванию образов для случая нестационарной генеральной совокупности.

Текст работы состоит из семи основных разделов. Первый раздел представленной работы описывает задачу обучения распознаванию образов в нестационарной генеральной совокупности, второй содержит краткую

характеристику существующих методов для задач распознавания указанного типа. Постановка задачи обучения распознаванию образов в нестационарной генеральной совокупности представлена в третьей части. Четвертая часть посвящена методу распознавания образов в нестационарной генеральной совокупности, построенному с использованием модели опорных векторов. Указанный метод был исследован в рамках бакалаврской работы[22]. В последующем разделе представлено описание классического метода логистической регрессии. Построение метода обучения распознаванию в нестационарной генеральной совокупности на основе модели логистической регрессии содержится в шестой части. В последнем седьмом разделе описаны результаты экспериментального исследования построенного алгоритма распознавания образов в нестационарной генеральной совокупности.

## **1. Проблема обучения распознаванию образов в условиях нестационарной генеральной совокупности**

Обычно в задачах распознавания образов предполагается, что свойства генеральной совокупности неизменны на протяжении всего процесса обучения. Однако мы можем столкнуться с задачами иного рода, в которых влияние каких-то скрытых факторов может привести к большим или меньшим изменениям в генеральной совокупности и, как следствие, в решающем правиле. Для такой ситуации обычно используются термины «нестационарная генеральная совокупность» («non-stationary environment») или «смещение концепта» («concept drift»).

Под смещением здесь понимается изменение свойств анализируемого явления, вследствие чего происходит «дрейф» решающего правила в признаковом пространстве. Поскольку единственным источником информации об исследуемом явлении являются объекты обучения, необходимо постоянное пополнение обучающего множества объектами, содержащими наиболее адекватные на данный момент данные о состоянии генеральной совокупности, что означает практически постоянный рост размера обучающей выборки. Возникает задача инкрементного обучения, когда после завершения построения решающего правила по заданному обучающему множеству в распоряжение разработчика поступают дополнительные объекты с известной для них скрытой характеристикой, которые было бы желательно использовать для дополнительного обучения, т.е. коррекции уже созданного классификатора. Однако, в случае нестационарной генеральной совокупности построенный в следующий момент времени классификатор может кардинально отличаться от предыдущего.

Таким образом, требуется по непрерывно поступающим объектам исследования осуществлять адаптацию решающего правила к происходящим изменениям в генеральной совокупности.

## 2. Постановка задачи обучения распознаванию образов в нестационарной генеральной совокупности

### 2.1. Классическая задача обучения распознаванию образов

В современной информатике интенсивно и успешно развивается методология анализа данных, направленная на алгоритмизацию поиска эмпирических закономерностей  $y(\omega): \Omega \rightarrow Y$  во множествах объектов  $\omega \in \Omega$ , вообще говоря, произвольной природы. Требуется, анализируя предъявленный массив данных, представляющий собой совокупность значений некоторых характеристик объектов  $(x(\omega_j), y(\omega_j))$  в пределах их доступного подмножества (обучающей выборки)  $\Omega^* = \{\omega_j, j = 1, \dots, N\} \subset \Omega$ , продолжить наблюдаемую связь между этими характеристиками на все гипотетическое множество объектов  $\omega \in \Omega$  (генеральную совокупность), чтобы можно было в дальнейшем для новых объектов, не участвовавших в обучении  $\omega \in \Omega \setminus \Omega^*$ , оценивать значения одних (целевых) характеристик  $y(\omega)$  через значения других характеристик  $x(\omega)$ , более доступных для непосредственного измерения:  $\hat{y}(x(\omega))$  [1,2,3,4].

В частности, если целевая характеристика принимает значения из конечного неупорядоченного множества  $Y = \{y^{(1)}, \dots, y^{(m)}\}$ , то такую задачу принято называть задачей обучения распознаванию образов. В простейшем случае, если рассматриваются лишь два класса, то в качестве двухэлементного множества их индексов обычно принимают множество  $Y = \{-1, 1\}$ .

Ограничимся рассмотрением задачи обучения распознаванию двух классов объектов, наблюдаемых через значения конечного числа их действительных признаков  $x_i(\omega)$ ,  $i = 1, \dots, n$ . В этом случае каждый объект генеральной совокупности  $\omega \in \Omega$  представлен точкой в конечномерном линейном пространстве признаков  $\mathbf{x}(\omega) = (x_1(\omega), \dots, x_n(\omega)) \in \mathbb{R}^n$ , а его скрытая фактическая принадлежность к одному из двух классов определяется



значением индекса класса  $y(\omega) \in \{1, -1\}$ . Тогда обучающая выборка  $\Omega^*$  примет вид  $\{\mathbf{X}, \mathbf{Y}\}$ , где  $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$  и  $\mathbf{Y} = \{y_i\}_{i=1}^N$ . Будем строить линейный классификатор:

$$a(\mathbf{x}) = \text{sign}(\mathbf{a}^T \mathbf{x} + b) \quad (1)$$

Классический подход к обучению распознаванию двух классов объектов, развитый В.Н. Вапником [4], основан на понимании модели генеральной совокупности в виде разделяющей гиперплоскости  $\mathbf{a}^T \mathbf{x} + b = 0$  в пространстве признаков  $\mathbb{R}^n$ , определяемой направляющим вектором  $\mathbf{a} \in \mathbb{R}^n$  и параметром положения  $b \in \mathbb{R}$ , априори неизвестными наблюдателю. Предполагается, что соответствующая линейная дискриминантная функция  $f(\mathbf{x}) = \mathbf{a}^T \mathbf{x} + b$  принимает в основном положительные значения для объектов первого класса и отрицательные – для второго:

$$f(\mathbf{x}(\omega)) = \mathbf{a}^T \mathbf{x}(\omega) + b \begin{cases} \text{преимущественно} > 0, & \text{если } y(\omega) = 1, \\ \text{преимущественно} < 0, & \text{если } y(\omega) = -1. \end{cases} \quad (2)$$

Обучающая совокупность  $\Omega^* = \{\omega_j, j = 1, \dots, N\} \subset \Omega$ , содержащая примеры объектов обоих классов, определяет два конечных множества точек в  $\mathbb{R}^n$ :

$$\Omega^* = \{\omega_j, j = 1, \dots, N\} \subset \Omega, \quad \mathbf{x}_j = \mathbf{x}(\omega_j) \in \mathbb{R}^n, \quad y_j = y(\omega_j) \in \{1, -1\}. \quad (3)$$

Требуется по обучающему множеству определить значения параметров  $\mathbf{a}$  и  $b$  для решающего правила (1).

## 2.2. Задача обучения распознаванию образов в нестационарной генеральной совокупности

Мы остаемся в рамках линейного подхода к распознаванию, и предполагаем, что основное свойство нестационарной генеральной совокупности выражается изменяющейся во времени разделяющей гиперплоскостью, характеризующей преимущественное различие векторов признаков объектов двух классов.

Изменяющаяся во времени разделяющая гиперплоскость полностью определяется своим направляющим вектором и параметром положения,

которые должны рассматриваться как функции времени  $\mathbf{a}_t: T \rightarrow \mathbb{R}^n$  и  $b_t: T \rightarrow \mathbb{R}$ , где  $T$  - упорядоченное множество моментов времени. Соответственно, всякий объект  $\omega \in \Omega$  рассматривается всегда только вместе с указанием момента времени  $t \in T$ , в который он предъявлен  $(\omega, t)$ :

$$f_t(\mathbf{x}(\omega)) = \mathbf{a}_t^T \mathbf{x}(\omega) + b_t \begin{cases} > 0, & y(\omega_t) = 1, \\ < 0, & y(\omega_t) = -1. \end{cases}$$

В классической постановке задачи обучения распознаванию образов каждый объект обучающей совокупности  $\{\omega_j, j = 1, \dots, N\}$  представлен вектором его признаков и индексом принадлежности определенному классу, так что выборка в целом является множеством пар  $\{(\mathbf{x}_j \in \mathbb{R}^n, y_j = \pm 1), j = 1, \dots, N\}$

В нестационарной интерпретации задачи обучения распознаванию образов каждый объект дополнительно характеризуется также моментом времени, в который был измерен его вектор признаков. В результате обучающая совокупность приобретает структуру множества троек  $\{(\mathbf{x}_t \in \mathbb{R}^n, y_t = \pm 1, D_t), t = 1, \dots, T\}$ , а учитывая возможность поступления в момент времени  $D_t$  нескольких объектов:

$\{(\mathbf{x}_j \in \mathbb{R}^n, y_j = \pm 1, D_t), j = (N_{t-1} + 1), K, N_t, t = 1, \dots, T, N_0 = 0\}$ , где  $N_t$  - индекс объекта, поступившего первым в соответствующий момент времени.

Согласно принятой концепции нестационарной генеральной совокупности в разные моменты времени  $D_t$  из числа моментов, определяемых обучающей последовательностью, скрытая от наблюдателя разделяющая гиперплоскость характеризуется разными неизвестными значениями направляющего вектора  $\mathbf{a}_t = \mathbf{a}_{D_t} \in \mathbb{R}^n$  и параметра положения  $b_t = b_{D_t} \in \mathbb{R}$ . Таким образом, объективно существует двухкомпонентный временной ряд со скрытой и наблюдаемой компонентами:

время	$D_1$	$D_2$	...	$D_t$	$D_{t+1}$	...	$D_{T-1}$	$D_T$
параметры гиперплоскости	$(a_1, b_1)$	$(a_2, b_2)$	...	$(a_t, b_t)$	$(a_{t+1}, b_{t+1})$	...	$(a_{T-1}, b_{T-1})$	$(a_T, b_T)$
вектора признаков и индексы класса	$(x_j, y_j)_{j=1}^{N_1}$	$(x_j, y_j)_{j=N_1+1}^{N_2}$	...	$(x_j, y_j)_{j=N_{t-1}+1}^{N_t}$	$(x_j, y_j)_{j=N_t+1}^{N_{t+1}}$	...	$(x_j, y_j)_{j=N_{T-1}+1}^{N_T}$	$(x_j, y_j)_{j=N_T+1}^{N_{T+1}}$

В классической постановке задачи распознавания образов нет свободы для интерпретации задачи обучения – требуется, анализируя предъявленную обучающую совокупность  $\{(x_j, y_j), j=1, \dots, N\}$ , дать оценку параметров разделяющей гиперплоскости  $(a, b)$ .

В рассматриваемой нами динамической постановке, задача обучения превращается в задачу анализа многокомпонентного временного ряда, в котором требуется, анализируя наблюдаемую компоненту, дать оценку скрытой компоненты. Это стандартная задача анализа временных рядов, специфика которой заключается лишь в предполагаемой модели связи между скрытой и наблюдаемой компонентами сигнала.

Согласно классификации задач оценивания скрытой компоненты сигнала, введенной Норбертом Винером [5], мы будем различать три основных вида задач обучения [6].

**Задача фильтрации обучающей последовательности.** Пусть  $D_t$  – момент поступления очередного объекта, к которому уже зарегистрированы векторы признаков и индексы классов объектов

$$\left\{ \dots, \left\{ (x_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (x_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \left\{ (x_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right\},$$

поступивших в предыдущие моменты времени до текущего момента включительно  $(\dots, D_{t-2}, D_{t-1}, D_t)$ . Требуется, анализируя уже поступившую часть обучающей последовательности, дать оценку параметров разделяющей гиперплоскости в текущий момент времени:

$$(\hat{\mathbf{a}}_{N_t|N_t}, \hat{\mathbf{b}}_{N_t|N_t}) = (\hat{\mathbf{a}}_{N_{D_t}|N_{D_t}}, \hat{\mathbf{b}}_{N_{D_t}|N_{D_t}}) =$$

$$= F_t \left( \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right)$$

Здесь двойная индексация оценки  $N_t | N_t$  указывает на то, что ищется оценка параметров разделяющей гиперплоскости в момент поступления  $N_t$ -го объекта после получения ровно  $N_t$  элементов обучающей последовательности.

**Задача интерполяции обучающей последовательности.** Пусть, по-прежнему, обучающая последовательность зарегистрирована вплоть до некоторого момента времени  $D_t$ , так что доступна последовательность

$$\left\{ \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right\}.$$

Требуется оценить параметры разделяющей гиперплоскости в некоторый предыдущий момент времени  $D_k$ ,  $k < t$ :

$$(\hat{\mathbf{a}}_{N_k|N_t}, \hat{\mathbf{b}}_{N_k|N_t}) = (\hat{\mathbf{a}}_{N_{D_k}|N_{D_t}}, \hat{\mathbf{b}}_{N_{D_k}|N_{D_t}}) =$$

$$= F_t \left( \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{k-1} + 1), K, N_k \right\}, \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right)$$

Двойная индексация  $N_k | N_t$  подчеркивает, что речь идет об оценке параметров разделяющей гиперплоскости в момент поступления  $N_k$ -го объекта после того, как зарегистрирована вся обучающая последовательность длины  $N_t$ .

**Задача экстраполяции обучающей последовательности (задача прогноза).** Анализируя часть обучающей последовательности, поступившую до некоторого момента времени  $D_t$  включительно, требуется оценить значения параметров разделяющей гиперплоскости в некоторый будущий момент времени  $\tau > D_t$ :

$$(\hat{\mathbf{a}}_{N_{\tau}|N_t}, \hat{\mathbf{b}}_{N_{\tau}|N_t}) = (\hat{\mathbf{a}}_{N_{D_t}|N_{D_t}}, \hat{\mathbf{b}}_{N_{D_t}|N_{D_t}}) =$$

$$= F_t \left( \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right).$$

### 2.3. Вероятностное описание нестационарности параметров решающего правила

Будем полагать, что в нулевой момент времени априорная плотность распределения параметров разделяющей гиперплоскости является равномерной и равно единице на всей числовой оси. Поскольку ее интеграл не равен единице, плотностью вероятности в классическом понимании она не является. Однако в [ 7 ] представлена возможность интерпретации такой функции как распределений вероятности. Подобные распределения называются несобственными.

В [ 8 ] предлагается для учета динамики решающего правила направляющий вектор  $\mathbf{a}_j \in \mathbb{R}^n$  рассматривать как случайный стационарный процесс:

$$\mathbf{a}_t = q\mathbf{a}_{t-1} + \xi_t, M(\xi_t) = \mathbf{0}, M(\xi_t \xi_t^T) = d\mathbf{I}, 0 \leq q < 1 \quad (4)$$

В силу стационарности процесса  $\mathbf{a}_j$  имеем  $M(\mathbf{a}_j) = \mathbf{0}$ ,  
 $M(\mathbf{a}_j \mathbf{a}_j^T) = M(\mathbf{a}_{j-1} \mathbf{a}_{j-1}^T) = M(\mathbf{a}_0 \mathbf{a}_0^T) = \mathbf{I}$

Справедлива следующая

**Теорема 1** [ 9 ]. Для случайного стационарного процесса (4) справедливо соотношение  $q = \sqrt{1-d}$ .

#### Доказательство

Т.к. случайные величины  $\mathbf{a}_j$ ,  $\xi_j$  независимы, то справедливо следующее соотношение:

$$M(\mathbf{a}_j \mathbf{a}_j^T) = q^2 M(\mathbf{a}_{j-1} \mathbf{a}_{j-1}^T) + M(\xi_j \xi_j^T)$$

Подставляем в это выражение следующие соотношения:  $M(\xi_j \xi_j^T) = d\mathbf{I}$ ,

$M(\mathbf{a}_j \mathbf{a}_j^T) = M(\mathbf{a}_{j-1} \mathbf{a}_{j-1}^T) = M(\mathbf{a}_0 \mathbf{a}_0^T) = \mathbf{I}$ , тогда получим

$$\begin{aligned}(1-q^2)\mathbf{I} &= d\mathbf{I} \\ q^2 &= 1-d \\ q &= \sqrt{1-d}\end{aligned}$$

Что и требовалось доказать.

Таким образом, априорное распределение параметра  $\mathbf{a}_t$  :

$$\psi_a(\mathbf{a}_t | \mathbf{a}_{t-1}) = N(\mathbf{a}_t | \sqrt{1-d}\mathbf{a}_{t-1}, d\mathbf{I})$$

Относительно параметра  $b_t$  аналогично предполагаем, что он является случайным стационарным процессом:

$$\begin{aligned}b_t &= b_{t-1} + \eta_t, M(\eta_t) = 0, M(\eta_t^2) = d' \\ \psi_b(b_t) &= N(b_t | b_{t-1}, d')\end{aligned}$$

Таким образом, совместная априорная плотность распределения параметров  $\mathbf{a}_t$  и  $b_t$  имеет вид:

$$\begin{aligned}\psi(\mathbf{a}_t, b_t | \mathbf{a}_{t-1}, b_{t-1}) &= N(\mathbf{a}_t | \sqrt{1-d}\mathbf{a}_{t-1}, d\mathbf{I})N(b_t | b_{t-1}, d') = \\ &= \frac{1}{d^{n/2}(2\pi)^{n/2}} \exp\left(-\frac{1}{2d}(\mathbf{a}_t - \sqrt{1-d}\mathbf{a}_{t-1})^T(\mathbf{a}_{t-1} - \sqrt{1-d}\mathbf{a}_{t-1})\right) \frac{1}{2d'} \exp\left(-\frac{1}{2d'}(b_t - b_{t-1})^2\right) \quad (5)\end{aligned}$$

### 3. Существующие методы обучения распознаванию образов в условиях нестационарной генеральной совокупности

Также как и при решении задач в условиях стационарной генеральной совокупности существующие методы распознавания для нестационарного случая можно разделить на те, которые используют одиночный классификатор, и те, что построены как ансамбль (композиция) классификаторов.

Большинство алгоритмов первой группы тем или иным образом эксплуатируют технологию временного окна, которая состоит в отборе из всех поступивших обучающих данных некоторого числа объектов, полученных непосредственно перед моментом контроля, предполагая, что именно в них содержится наиболее релевантная информация. Количество выбираемых объектов составляет длину окна, которая может быть постоянной или переменной (Рисунок 1).

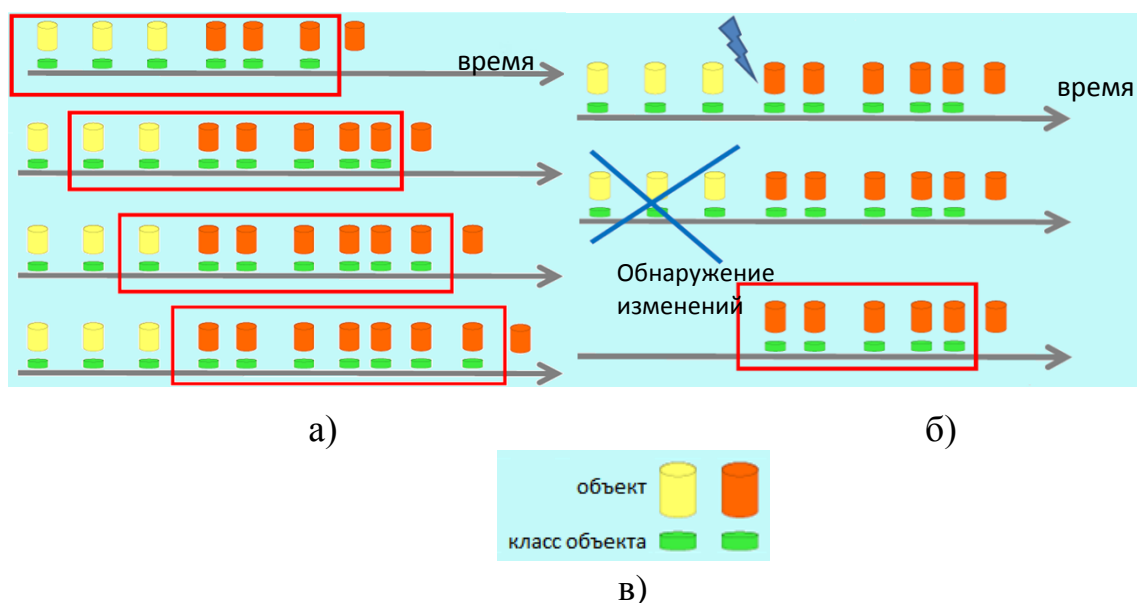


Рисунок 1 – Технология временного окна: (а) - постоянной длины; (б) – переменной длины, где объекты обозначены как (в)

Постоянная длина окна задается пользователем на основе каких-то априорных соображений о скорости изменений в среде, в случае же переменной длины для ее определения используется обнаружение

происходящих в генеральной совокупности изменений. Обнаружение может быть построено с помощью статистических методов [ 10 ]или теории информации [ 11 ].Наиболее известными методами этой группы являются алгоритмы семейства FLORA [12], ADWIN [13], а также TMF [14].

Отдельно стоит отметить алгоритмы обучения с одиночным классификатором, в которых временное окно не используется. В таких методах в процессе получения объектов обучающей выборки для обучения классификатора отбираются экстремальные объекты, содержащие наиболее важную и свежую информацию [ 15 ] или, наоборот, обнаруживаются и удаляются те объекты, которые содержат более не актуальную информацию, а обучение нового классификатора производится на оставшихся данных [16].

Более сложными и, как правило, более точными являются методы, основанные на композиции алгоритмов. В этом случае на множестве всех объектов обучающей выборки строится семейство решающих правил, которые объединяются затем на основе голосования или взвешенного голосования составляющих. Здесь можно выделить два подхода:

1) При поступлении новых данных выбросить из ансамбля худший по качеству классификатор, дополнив композицию новым, обученным на поступивших данных. Одним из первых алгоритмов этой группы был SEA [17]

2) При поступлении новых данных переобучить на них все классификаторы композиции.

Представителем этого подхода является AccuracyWeightedEnsemble (AWE) [18]



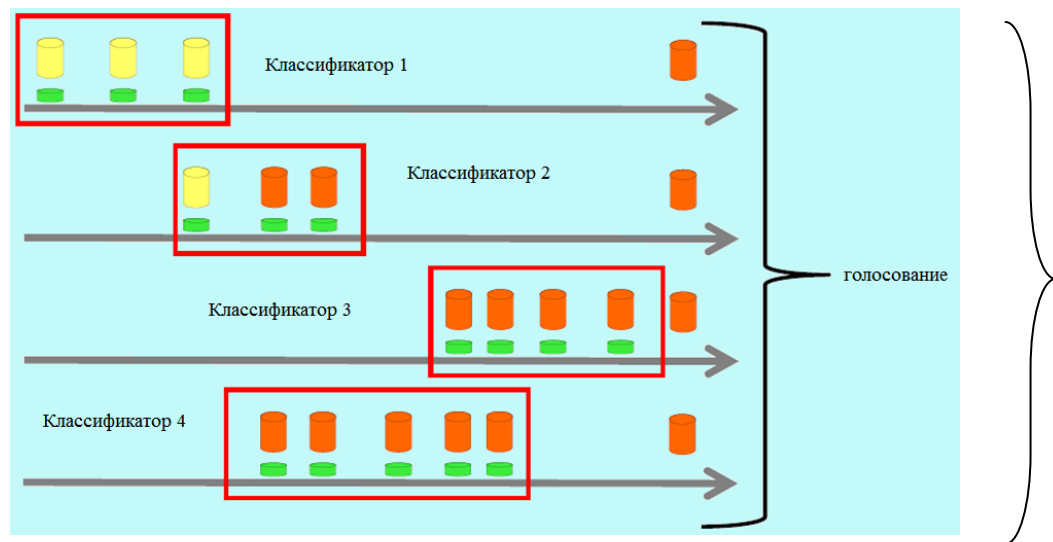


Рисунок 2 – Композиция классификаторов

В качестве составляющих композиции могут выбираться методы из первой группы, так построен, например ADWIN Bagging [19]. Как следует из названия, он использует алгоритм адаптивного окна ADWIN. Также компонентами ансамбля могут быть решающие деревья (в частности, большой популярностью пользуется их специальный вид - деревья Хефдинга), примером может служить EnsembleCombiningRestrictedHoeffdingTrees [20].

По сравнению со стационарным случаем, ситуация нестационарной генеральной совокупности изучена достаточно слабо, что объясняется относительной новизной рассматриваемой темы. В данной области отсутствует четкая система определений, имеют место разночтения в значениях терминов у различных авторов. Существующие методы являются в большей степени эвристическими и предназначены для решения одной конкретной задачи или узкой группы задач.

На первый взгляд это объясняется отсутствием острой необходимости в таких методах из-за небольшого количества задач именно для нестационарной генеральной совокупности, а также тем, что имеются методы распознавания для стационарного решающего правила, которые позволяют получить приемлемые результаты даже в случае небольшого его смещения. Однако, необходимо заметить, что ни один объект реального мира не

является неизменным во времени, отсюда следует, что учитывать поступающие объекты необходимо в неразрывной связи с тем моментом времени, в который они были получены. Таким образом, любая задача распознавания является задачей в нестационарной генеральной совокупности и должна рассматриваться именно так.

Тогда очевидно, что использование для их решения алгоритмов, учитывающих смещение решающего правила, будет гораздо более эффективно, нежели алгоритмов, этого не учитывающих. Особенно четко это видно в тех ситуациях, в которых пренебрежение смещением решающего правила может привести к недопустимо большой ошибке при распознавании.

## 4. Обобщение метода опорных векторов для нестационарной генеральной совокупности

### 4.1. Метод опорных векторов в классической задаче обучения распознаванию образов

Естественно в качестве оценки модели генеральной совокупности попытаться выбрать параметры линейной дискриминантной функции так, чтобы она принимала положительные значения для объектов первого класса и отрицательные для другого:  $y_j(\mathbf{a}^T \mathbf{x}_j + b) > 0$ . Если такие значения параметров найдутся (обучающая совокупность линейно разделима, т.е. выпуклые оболочки множеств точек двух классов не пересекаются), то существует множество значений, удовлетворяющих этим неравенствам. Выдвинутая В.Н. Вапником концепция оптимальной разделяющей гиперплоскости [4] заключается в выборе таких значений параметров, при которых «зазор» между минимальным положительным и максимальным отрицательным значениями дискриминантной функции был бы максимальным:  $\varepsilon \rightarrow \max$  при условиях  $\mathbf{a}^T \mathbf{x}_j + b \geq \varepsilon > 0$  для объектов первого класса  $y_j = 1$  и  $\mathbf{a}^T \mathbf{x}_j + b \leq -\varepsilon < 0$  для объектов второго класса  $y_j = -1$ , или, что эквивалентно,  $y_j(\mathbf{a}^T \mathbf{x}_j + b) \geq \varepsilon > 0$  для всех объектов обучающей совокупности  $j = 1, \dots, N$ .

Здесь евклидова норма направляющего вектора  $\|\mathbf{a}\| = (\mathbf{a}^T \mathbf{a})^{1/2}$  условна, ее всегда можно принять равной единице, изменив величину параметра положения  $b$ . Буквальное понимание концепции оптимальной разделяющей гиперплоскости приводит к следующему критерию обучения:

$$\begin{cases} \varepsilon \rightarrow \max, \\ y_j(\mathbf{a}^T \mathbf{x}_j + b) \geq \varepsilon > 0, \quad j = 1, \dots, N, \\ \mathbf{a}^T \mathbf{a} = 1. \end{cases} \quad (6)$$

Обычно этот критерий записывают в более удобной форме, которая получается из (6) делением обеих частей всех неравенств на  $\varepsilon$

$$y_j \left[ \left( (1/\varepsilon) \mathbf{a} \right)^T \mathbf{x}_j + (1/\varepsilon) b \right] \geq 1$$

и введением обозначений  $(1/\varepsilon) \mathbf{a} = \mathbf{a}^*$  и  $(1/\varepsilon) b = b^*$ . В этих обозначениях  $\mathbf{a}^{*T} \mathbf{a}^* = (1/\varepsilon^2) \mathbf{a}^T \mathbf{a}$ , т.е.  $\mathbf{a}^{*T} \mathbf{a}^* = (1/\varepsilon^2)$  при выполнении ограничения  $\mathbf{a}^T \mathbf{a} = 1$ , требование максимизации «зазора»  $\varepsilon \rightarrow \max$  эквивалентно требованию минимизации квадрата нормы направляющего вектора  $\mathbf{a}^{*T} \mathbf{a}^* \rightarrow \min$ . Сохраняя прежние обозначения  $\mathbf{a}$  и  $b$  для масштабированных параметров, мы получим эквивалентную запись критерия обучения (6):

$$\begin{cases} \mathbf{a}^T \mathbf{a} \rightarrow \min, \\ y_j (\mathbf{a}^T \mathbf{x}_j + b) \geq \varepsilon > 0, \quad j = 1, \dots, N. \end{cases}$$

Если выпуклые оболочки двух классов в обучающей совокупности пересекаются, В.Н. Вапник предложил использовать следующий критерий, вообще говоря, эвристический, допускающий насильственное смещение точек, «мешающих» линейному разделению:

$$\begin{cases} J(\mathbf{a}, b, \delta_j, j = 1, \dots, N) = \mathbf{a}^T \mathbf{a} + C \sum_{j=1}^N \delta_j \rightarrow \min, \\ y_j (\mathbf{a}^T \mathbf{x}_j + b) \geq 1 - \delta_j, \quad \delta_j \geq 0, \quad j = 1, \dots, N. \end{cases} \quad (7)$$

Очевидно, что это задача квадратичного программирования, т.е. задача минимизации квадратичной функции при ограничениях и виде линейных неравенств. Заметим, что каждое ограничение соответствует одному объекту обучающей совокупности, требуя, чтобы этот объект, по-возможности, правильно классифицировался дискриминантной функцией.

Именно этот критерий обучения распознаванию двух классов объектов получил огромную популярность в литературе под названием метода опорных векторов. Такое название определяется тем обстоятельством, что в точке минимума целевой функции активными оказываются лишь небольшое число ограничений-неравенств, соответствующих векторам признаков лишь некоторых объектов, которые полностью определяют оптимальные значения параметров разделяющей гиперплоскости. Эти векторы называются

опорными векторами в данной обучающей совокупности, давая название методу в целом.

Задачу квадратичного программирования (7) обычно записывают и решают в двойственной форме, в которой роль опорных векторов становится очевидной.

#### 4.2. Вероятностное обоснование метода опорных векторов

Пусть в пространстве признаков  $R^n$  объективно определена разделяющая гиперплоскость  $\mathbf{a}^T \mathbf{x} + b = 0$ , неизвестная наблюдателю. В качестве модели генеральной совокупности будем рассматривать два параметрических семейства плотностей распределения вероятностей,  $\varphi_1(\mathbf{x}(\omega) | \mathbf{a}, b)$  и  $\varphi_{-1}(\mathbf{x}(\omega) | \mathbf{a}, b)$ , связанных с двумя классами объектов  $y(\omega) = 1$  и  $y(\omega) = -1$ , и сконцентрированные преимущественно по разные стороны гиперплоскости. Совместную плотность распределения конечного множества векторов признаков объектов известных классов в составе обучающей совокупности (3) будем понимать как плотность распределения выборки независимых реализаций этих двух распределений:

$$\Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}, b) = \prod_{j=1}^N \varphi_{y_j}(\mathbf{x}_j | \mathbf{a}, b) = \left( \prod_{j: y_j=1} \varphi_1(\mathbf{x}_j | \mathbf{a}, b) \right) \left( \prod_{j: y_j=-1} \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b) \right).$$

Пусть, далее, выбрана априорная плотность совместного распределения вероятностей  $\psi(\mathbf{a}, b)$  для параметров распределений  $\varphi_1(\mathbf{x}(\omega) | \mathbf{a}, b)$  и  $\varphi_{-1}(\mathbf{x}(\omega) | \mathbf{a}, b)$ . Тогда апостериорная плотность распределения параметров  $\mathbf{a}$  и  $b$  относительно обучающей совокупности определяется формулой Байеса:

$$p(\mathbf{a}, b | y_j, j = 1, \dots, N) = \frac{\psi(\mathbf{a}, b) \Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}, b)}{\int \psi(\mathbf{a}', b') \Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}', b') d\mathbf{a}' db'}. \quad (8)$$

Поскольку знаменатель не зависит от целевых переменных, то достаточно рассматривать только числитель:

$$p(\mathbf{a}, b | y_j, j = 1, \dots, N) \propto \psi(\mathbf{a}, b) \Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}, b) =$$

$$\psi(\mathbf{a}, b) \left( \prod_{j: y_j=1} \varphi_1(\mathbf{x}_j | \mathbf{a}, b) \right) \left( \prod_{j: y_j=-1} \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b) \right).$$

Принцип максимизации плотности апостериорного распределения в пространстве параметров модели генеральной совокупности приводит к байесовскому правилу обучения:

$$\left( \hat{\mathbf{a}}, \hat{b} | (\mathbf{x}_j, y_j), j = 1, \dots, N \right) = \arg \max_{\mathbf{a}, b} p(\mathbf{a}, b | y_j, j = 1, \dots, N) =$$

$$\arg \max_{\mathbf{a}, b} \left[ \log \psi(\mathbf{a}, b) + \sum_{j: y_j=1} \log \varphi_1(\mathbf{x}_j | \mathbf{a}, b) + \sum_{j: y_j=-1} \log \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b) \right]. \quad (9)$$

В работе [21] предложена следующая вероятностная модель генеральной совокупности. Рассмотрим несобственные плотности распределения  $\varphi_1(\mathbf{x}(\omega) | \mathbf{a}, b)$  и  $\varphi_{-1}(\mathbf{x}(\omega) | \mathbf{a}, b)$ , определяемые выражениями

$$\varphi_1(\mathbf{x} | \mathbf{a}, b) = \begin{cases} 1, & \mathbf{a}^T \mathbf{x} + b > 1, \\ \exp \left[ -c \left( 1 - (\mathbf{a}^T \mathbf{x} + b) \right) \right], & \mathbf{a}^T \mathbf{x} + b < 1, \end{cases}$$

$$\varphi_{-1}(\mathbf{x} | \mathbf{a}, b) = \begin{cases} 1, & \mathbf{a}^T \mathbf{x} + b < -1, \\ \exp \left[ -c \left( 1 + (\mathbf{a}^T \mathbf{x} + b) \right) \right], & \mathbf{a}^T \mathbf{x} + b > -1. \end{cases} \quad (10)$$

Наглядное представление этой пары несобственных плотностей распределения приведено на Рисунке 3. Очевидно, что распределения полностью соответствуют качественной модели генеральной совокупности(2), являясь равномерными при достаточном удалении от разделяющей гиперплоскости  $\mathbf{a}^T \mathbf{x} + b > 1$  либо  $\mathbf{a}^T \mathbf{x} + b < -1$ , а также вдоль любой прямой, параллельной ей. Конкретизация качественной модели (2) заключается в том, что несобственные плотности приняты экспоненциально уменьшающимися при  $\mathbf{a}^T \mathbf{x} + b < 1$  и, соответственно,  $\mathbf{a}^T \mathbf{x} + b > -1$ .

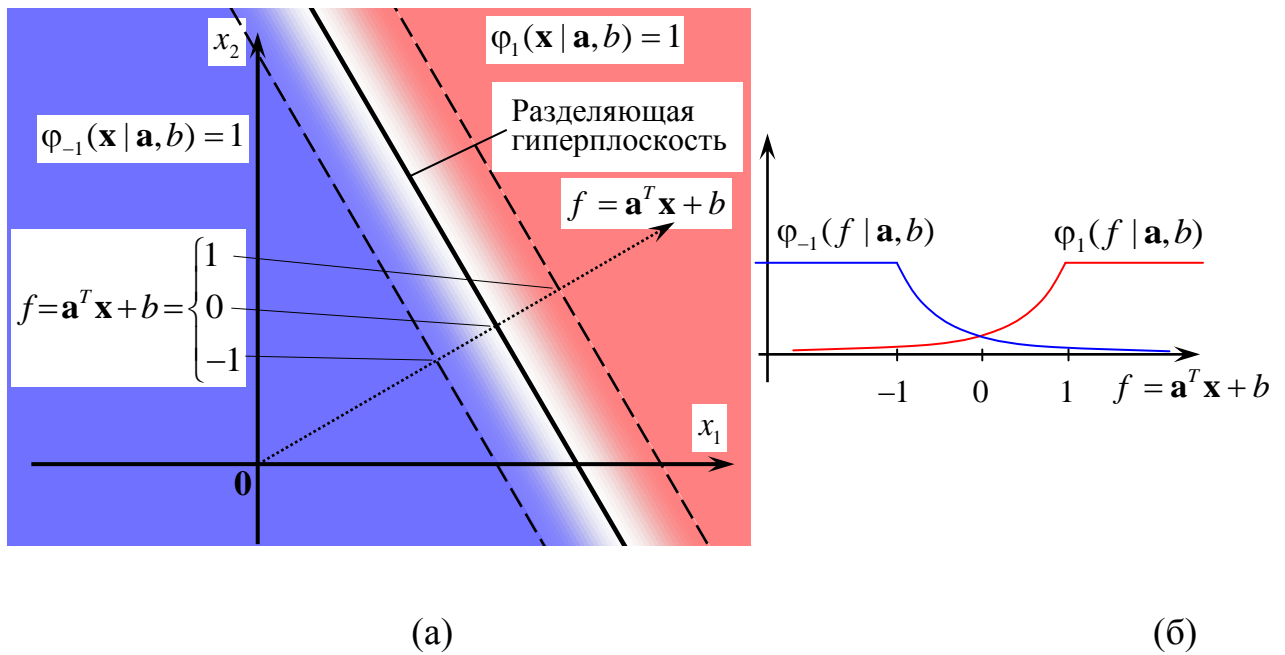


Рисунок 3 - Яркое представление несобственных плотностей распределения двух классов в двумерном пространстве признаков (а) и их значения вдоль направляющего вектора разделяющей гиперплоскости (б).

Направляющий вектор разделяющей гиперплоскости  $\mathbf{a}$  будем считать априори нормально распределенным с независимыми компонентами, имеющими одинаковые априорные дисперсии  $\sigma^2$ . Что же касается параметра положения разделяющей гиперплоскости  $b$ , то будем считать, что отсутствуют какие-либо априорные предположения о его значении, что выражается равномерным несобственным распределением, равным единице на всей числовой оси. Тогда совместное априорное распределение параметров разделяющей гиперплоскости будет выражаться плотностью

$$\psi(\mathbf{a}, b | \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2} \mathbf{a}^T \mathbf{a}\right).$$

Очевидно, что это несобственная плотность, поскольку ее интеграл по переменной  $b$  не существует.

Для несобственных плотностей распределения векторов признаков объектов двух классов и априорного несобственного распределения

параметров разделяющей гиперплоскости байесовский критерий обучения(9) примет вид[21]

$$\begin{aligned} & \left( \hat{\mathbf{a}}, \hat{b} \mid (\mathbf{x}_j, y_j), j = 1, \dots, N \right) = \\ & \arg \min_{\mathbf{a}, b} \left[ \frac{1}{2\sigma^2} \mathbf{a}^T \mathbf{a} + c \sum_{\substack{j: y_j=1 \\ \mathbf{a}^T \mathbf{x}_j + b < 1}} (1 - (\mathbf{a}^T \mathbf{x}_j + b)) + c \sum_{\substack{j: y_j=-1 \\ \mathbf{a}^T \mathbf{x}_j + b > -1}} (1 + (\mathbf{a}^T \mathbf{x}_j + b)) \right]. \end{aligned} \quad (11)$$

При выборе коэффициента штрафа на сумму смещений точек обучающей совокупности  $C = 2\sigma^2 c$  этот критерий полностью идентичен классическому критерию обучения методом опорных векторов.

#### 4.3. Распространение метода опорных векторов на случай нестационарной генеральной совокупности

Вспомним описанное в пункте 2.2 предположение, заключающееся в описании свойства нестационарности генеральной совокупности при помощи разделяющей гиперплоскости, со временем изменяющей свое положение. Следовательно, ее направляющий вектор и параметр положения рассматриваются как функции времени  $\mathbf{a}_t$  и  $b_t$ :  $f_t(\mathbf{x}(\omega)) = \mathbf{a}_t^T \mathbf{x} + b_t$  и модельное представление генеральной совокупности в виде несобственных распределений двух классов в пространстве признаков(10) будет иметь вид:

$$\varphi(\mathbf{x} \mid \mathbf{a}_t, b_t, y; c) = \begin{cases} const, & yz(\mathbf{a}_t, \mathbf{x}) \geq 1, \\ e^{-c(1-yz(\mathbf{a}_t, \mathbf{x}))}, & yz(\mathbf{a}_t, \mathbf{x}) < 1, \end{cases}$$

где  $z(\mathbf{x}, \mathbf{a}_t) = \mathbf{a}_t^T \mathbf{x} + b = 0$ .

Тогда совместная условная априорная вероятность принадлежности объектов в составе обучающей совокупности выражается как произведение:

$$\Phi(\mathbf{Y} \mid \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2) = \prod_{j=1}^{N_T} \varphi(\mathbf{x} \mid \mathbf{a}_t, b_t, y; c)$$

Пусть совместная априорная плотность распределения параметров за весь период времени определяется как произведение плотностей для отдельных отсчетов:

$$\Psi(\mathbf{a}_t, b_t, t = 1, \dots, T) = \prod_{t=1}^T \psi_t(\mathbf{a}_t, b_t \mid \mathbf{a}_{t-1}, b_{t-1}),$$



где условная априорная плотность  $\psi(\mathbf{a}_t, b_t | \mathbf{a}_{t-1}, b_{t-1})$  определяется согласно (5). Будем предполагать, что в нулевой момент времени априорное распределение параметров разделяющей гиперплоскости является несобственным и имеет вид:

$$\psi_0(\mathbf{a}_0, b_0) \propto \psi_0(\mathbf{a}_0) = N(\mathbf{a}_0 | \mathbf{0}, \mathbf{I}).$$

Применение принципа максимизации апостериорной вероятности для оценивания последовательности параметров  $(\mathbf{a}_t, b_t)_{t=1}^T$  при принятых предположениях об априорных плотностях параметров направляющего вектора и плотностях классов приводит к следующему оптимизационному критерию

$$J(\mathbf{a}_t, b_t, \delta_t, t=0, \dots, T) = \mathbf{a}_0^T \mathbf{a}_0 + \frac{1}{d} \sum_{t=1}^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1})^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1}) + \frac{1}{d'} \sum_{t=1}^T (b_t - b_{t-1})^2 + \sum_{t=1}^T \sum_{j=1}^{N_t} \delta_{j,t} \rightarrow \min_{[\mathbf{a}_t, b_t]_{j=1}^T} \quad (12)$$

$$y_{j,t}(\mathbf{a}_t^T \mathbf{x}_{j,t} + b_t) \geq 1 - \delta_{j,t}, \delta_{j,t} \geq 0, \\ j = 1, \dots, N_t, t = 1, \dots, T$$

который мы перепишем в общем виде:

$$J(\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_T) = \sum_{t=0}^T [(\mathbf{z}'_t - \mathbf{z}^0_t)^T \mathbf{Q}_t (\mathbf{z}'_t - \mathbf{z}^0_t) + \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t] + \sum_{t=1}^T (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \rightarrow \min_{\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_T}, \\ \mathbf{g}_j^T \cdot \mathbf{z}'_t + z''_j - 1 \geq 0, j = (N_{t-1} + 1), \dots, N_t, t = 1, \dots, T, \\ z''_j \geq 0, j = (N_{t-1} + 1), \dots, N_t, t = 1, \dots, T. \quad (13)$$

где

$$\mathbf{z}_t = \begin{bmatrix} \mathbf{z}'_t \\ \mathbf{z}''_t \end{bmatrix}, \mathbf{z}'_t = \begin{bmatrix} \mathbf{a}_t \\ b_t \end{bmatrix}, \mathbf{z}''_t = [\delta_j, j = (N_{t-1} + 1), \dots, N_t], \mathbf{g}_j = \begin{bmatrix} y_j \mathbf{x}_j \\ y_j \end{bmatrix}, j = (N_{t-1} + 1), \dots, N_t, \mathbf{e}_t = [1]_{N_{t-1}+1}^{N_t}, \\ \mathbf{Q}_t^0 = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}, \mathbf{U}_t = \begin{bmatrix} \frac{1}{d} & 0 & \dots & 0 & 0 \\ 0 & \frac{1}{d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{d} & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{d'} \end{bmatrix}, \mathbf{A}_t = \begin{bmatrix} \sqrt{1-d} & 0 & \dots & 0 & 0 \\ 0 & \sqrt{1-d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \sqrt{1-d} & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \\ t = 1, \dots, T$$

#### 4.4. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода опорных векторов

В работе [ 22 ] был предложен алгоритм, основанный на идее использования для оптимизации получившегося парно-сепарабельного критерия (12) общего принципа динамического программирования.

Введем новые обозначения

$$\mathbf{z}_t = \begin{bmatrix} \mathbf{z}'_t & z''_t \end{bmatrix}^T, \quad \mathbf{z}'_t = \begin{bmatrix} \mathbf{a}_t^T & b_t \end{bmatrix}^T, \quad \mathbf{z}''_t = \begin{bmatrix} \delta_j \end{bmatrix}_{j=1}^{N_t}, \quad \zeta_t(\mathbf{z}'_t) = (\mathbf{z}'_t - \mathbf{z}_t^0)^T \mathbf{Q}_t^0 (\mathbf{z}'_t - \mathbf{z}_t^0),$$

$$\chi_t(\mathbf{z}''_t) = \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t, \quad \mathbf{e}_t = [1]_1^{N_t}, \quad t = 1, \dots, T, \quad \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) = (\mathbf{z}'_t - \mathbf{A}_j \mathbf{z}'_{t-1})^T \mathbf{U}_j (\mathbf{z}'_t - \mathbf{A}_j \mathbf{z}'_{t-1})$$

и перепишем критерий (12) в более удобном виде

$$J(\mathbf{z}_0, \dots, \mathbf{z}_T) = \sum_{t=0}^T \zeta_t(\mathbf{z}'_t) + \sum_{t=0}^T \chi(\mathbf{z}''_t) + \sum_{t=1}^T \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) \rightarrow \min, \quad \mathbf{z}_t \in Z_t$$

где области допустимых значений переменных определяются условиями

$$\{\mathbf{z} \in \square^{n+2} : \mathbf{g}_j^T \cdot \mathbf{z}'_t + z''_j - 1 \geq 0, \quad j = (N_{t-1} + 1), \dots, N_t, \quad t = 0, \dots, T, \quad \mathbf{z}''_t \geq 0\}.$$

Метод динамического программирования основан на понятии последовательности функций Беллмана

$$\tilde{J}_t(\mathbf{z}_t) = \min_{\mathbf{z}_0, \dots, \mathbf{z}_{t-1}} J_t([\mathbf{z}_s]_{s=1}^t), \quad [\mathbf{z}_s \in Z_s]_{s=0}^{t-1},$$

связанных с частичными критериями

$$J_t(\mathbf{z}_0, \dots, \mathbf{z}_t) = \sum_{s=0}^t \zeta_s(\mathbf{z}'_s) + \sum_{s=0}^t \chi(\mathbf{z}_s) + \sum_{s=1}^t \gamma_s(\mathbf{z}'_{s-1}, \mathbf{z}'_s),$$

имеющими такую же структуру, как и полная целевая функция, но определенными на множестве переменных  $Z_t = (\mathbf{z}_s, s = 0, \dots, t)$ . Более подробно о методе динамического программирования и функциях Беллмана будет сказано в следующем разделе. Для получения фильтрационных оценок параметров разделяющей гиперплоскости мы будем использовать фундаментальное свойство функции Беллмана

$$\tilde{J}_t(\mathbf{z}_t) = \zeta_t(\mathbf{z}'_t) + \chi(\mathbf{z}''_t) + \min_{\mathbf{z}'_{t-1}, \mathbf{z}''_{t-1}} \left[ \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) + \tilde{J}_{t-1}(\mathbf{z}_{t-1}) \right] \quad (14)$$

которое называется прямым рекуррентным соотношением. Процедура начинается со значения первой функции Беллмана  $\tilde{J}_0(\mathbf{z}_0) = \zeta_0(\mathbf{z}_0') + \chi(\mathbf{z}_0'')$ . Затем, функции Беллмана рекуррентно пересчитываются для последующих отсчетов, в соответствии с прямым рекуррентным соотношением. При этом минимум функции Беллмана на каждом шаге определяет фильтрационное значение параметров оптимальной разделяющей гиперплоскости

$$\hat{\mathbf{z}}_t = \arg \min_{\mathbf{z}_t} \hat{J}_t(\mathbf{z}_t), \mathbf{z}_t \in Z_t \quad (15)$$

#### 4.5. Численная реализация процедуры динамического программирования для оценивания параметров оптимальной разделяющей гиперплоскости

Предполагается, что существует подходящая компактная форма представления функций Беллмана, позволяющая хранить эти функции в памяти. В нашем случае предыдущая функция Беллмана в (14) является кусочно-квадратичной. При этом дробность кусочной квадратичности функций Беллмана будет нарастать на каждом шаге процедуры динамического программирования, что означает невозможность подобрать для них адекватное конечно-параметрическое семейство. Это делает невозможным численную реализацию процедуры динамического программирования.

Предложенная в [ 23 ] идея приближенной реализации процедуры динамического программирования заключается в замене функции

$$F_t(\mathbf{z}_t') = \min_{\mathbf{z}_{t-1} \in Z_{t-1}} [\gamma_t(\mathbf{z}_{t-1}', \mathbf{z}_t') + \tilde{J}_{t-1}(\mathbf{z}_{t-1})]$$

подходящей квадратичной функцией  $\hat{F}_t(\mathbf{z}_t') = \hat{c}_t + (\mathbf{z}_t' - \hat{\mathbf{z}}_t)^T \hat{\mathbf{Q}}_t (\mathbf{z}_t' - \hat{\mathbf{z}}_t)$

Тогда квадратичными будут и следующие аппроксимации функций Беллмана и станет возможной численная реализация процедуры динамического программирования.

Таким образом, квадратичная аппроксимация очередной функций Беллмана сводится к подбору подходящих значений параметров  $(\hat{c}_t, \hat{\mathbf{z}}_t, \hat{\mathbf{Q}}_t)$

квадратичной функции  $\hat{F}_t(\mathbf{z}'_t)$ , которые обеспечивали бы сохранение основных особенностей, вообще говоря, неквадратичной функции и, следовательно, исходной функции Беллмана. Такими особенностями являются, положение точек минимума функции  $\hat{\mathbf{z}}_t = \arg \min F_t(\mathbf{z}'_t)$ , значения в точках минимума  $\hat{c}_t = \min F_t(\mathbf{z}'_t)$ , а также матрица вторых производных в точке минимума  $\hat{\mathbf{Q}}_t = \nabla^2 F_t(\mathbf{z}'_t) \Big|_{\arg \min F_t(\mathbf{z}'_t)}$ . Представляется предпочтительным передать эти параметры в точности, выбрав

$$\hat{c}_t = \min F_t(\mathbf{z}'_t), \hat{\mathbf{z}}_t = \arg \min F_t(\mathbf{z}'_t), \hat{\mathbf{Q}}_t = \nabla^2 F_t(\mathbf{z}'_t) \Big|_{\arg \min F_t(\mathbf{z}'_t)} \quad (16)$$

**Теорема 2[22].** Пусть предыдущая функция Беллмана  $\bar{J}_{t-1}(\mathbf{z}_{t-1})$  является квадратичной по  $\mathbf{z}'_{t-1}$  и линейной по  $\mathbf{z}''_{t-1}$ , функция связи  $\gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t)$  является квадратичной.

$$\begin{aligned} \bar{J}_{t-1}(\mathbf{z}_{t-1}) &= \tilde{c}_{t-1} + (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}''_{t-1} \\ \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) &= (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \end{aligned}$$

Тогда параметры  $(\hat{c}_t, \hat{\mathbf{z}}_t, \hat{\mathbf{Q}}_t)$  квадратичной аппроксимации  $\hat{F}_t(\mathbf{z}'_t)$  функции  $F_t(\mathbf{z}'_t)$ , удовлетворяющие условиям (16), имеют вид

$$\begin{aligned} \hat{\mathbf{z}}_t &= \dot{\mathbf{z}}'_{t-1} \\ \dot{\mathbf{z}}_{t-1} &= \begin{bmatrix} \dot{\mathbf{z}}'_{t-1} \\ \dot{\mathbf{z}}''_{t-1} \end{bmatrix} \end{aligned} \quad (17)$$

$$\hat{c}_t = \tilde{c}_{t-1} + \mathbf{C} \mathbf{e}_{t-1}^T \dot{\mathbf{z}}''_{t-1} + (\dot{\mathbf{z}}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\dot{\mathbf{z}}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) \quad (18)$$

$$\hat{\mathbf{Q}}_t = \mathbf{U}_t^T \mathbf{A}_t (\mathbf{H}_{t-1}^{11})^T \bar{\mathbf{Q}}_{t-1} \mathbf{H}_{t-1}^{11} \mathbf{A}_t^T \mathbf{U}_t + (\mathbf{A}_t \mathbf{H}_{t-1}^{11} \mathbf{A}_t^T \mathbf{U}_t - \mathbf{I})^T \mathbf{U}_t (\mathbf{A}_t \mathbf{H}_{t-1}^{11} \mathbf{A}_t^T \mathbf{U}_t - \mathbf{I}) \quad (19)$$

где

$$(\dot{\mathbf{z}}_{t-1}) = \arg \min_{\substack{\mathbf{z}_{t-1}, \\ \mathbf{g}_j^T \mathbf{z}_{t-1} + z_j'' - 1 \geq 0, \\ j=(N_{t-2}+1), \dots, N_{t-1}, \\ \mathbf{z}_{t-1} \geq 0}} \left[ (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}''_{t-1} \right] \quad (20)$$

Матрица  $\mathbf{H}_{t-1}^{11}$  является левым верхним блоком  $(n \times n)$  матрицы

$$\mathbf{H}_{t-1} = \begin{pmatrix} \mathbf{H}_{t-1}^{11} & \mathbf{H}_{t-1}^{12} & \mathbf{H}_{t-1}^{13} & \mathbf{H}_{t-1}^{14} & \mathbf{H}_{t-1}^{15} \\ \mathbf{H}_{t-1}^{21} & \mathbf{H}_{t-1}^{22} & \mathbf{H}_{t-1}^{23} & \mathbf{H}_{t-1}^{24} & \mathbf{H}_{t-1}^{25} \\ \mathbf{H}_{t-1}^{31} & \mathbf{H}_{t-1}^{32} & \mathbf{H}_{t-1}^{33} & \mathbf{H}_{t-1}^{34} & \mathbf{H}_{t-1}^{35} \\ \mathbf{H}_{t-1}^{41} & \mathbf{H}_{t-1}^{42} & \mathbf{H}_{t-1}^{43} & \mathbf{H}_{t-1}^{44} & \mathbf{H}_{t-1}^{45} \\ \mathbf{H}_{t-1}^{51} & \mathbf{H}_{t-1}^{52} & \mathbf{H}_{t-1}^{53} & \mathbf{H}_{t-1}^{54} & \mathbf{H}_{t-1}^{55} \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{Q}}_{t-1} + \mathbf{A}_t^T \mathbf{U}_t \mathbf{A}_t & -\mathbf{G}'^T & \mathbf{0} & \mathbf{0} & -\mathbf{G}''^T \\ \mathbf{G}' & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{V}'^T & \mathbf{V}''^T \\ \mathbf{0} & \mathbf{0} & \mathbf{V}' & \mathbf{0} & \mathbf{0} \\ \mathbf{G}'' & \mathbf{0} & \mathbf{V}'' & \mathbf{0} & \mathbf{0} \end{pmatrix}^{-1},$$

где  $\mathbf{G}_{t-1} = [\mathbf{g}_j^T]_{j=N_{t-2}+1}^{N_{t-1}}$ .

**Доказательство.**

Введем обозначения

$$(\dot{\mathbf{z}}_{t-1}, \hat{\mathbf{z}}_t) = \arg \min_{\substack{\mathbf{z}_{t-1}, \mathbf{z}_t \\ \mathbf{g}_j^T \mathbf{z}'_{t-1} + \mathbf{z}_j'' - 1 \geq 0, j=(N_{t-2}+1), \dots, N_{t-1} \\ \mathbf{z}_{t-1}'' \geq 0}} \left[ \tilde{c}_{t-1} + (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}_{t-1}'' + \right. \\ \left. + (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \right]$$

Очевидно, что компонента  $\hat{\mathbf{z}}_t$  этого составного вектора является точкой минимума функции  $F_t(\mathbf{z}'_t)$ . Эта задача оптимизации распадается на две независимых задачи оптимизации по каждой из переменных  $\dot{\mathbf{z}}_{t-1}$  и  $\hat{\mathbf{z}}_t$  в отдельности,

$$\begin{aligned} \dot{\mathbf{z}}_{t-1} &= \arg \min_{\substack{\mathbf{z}_{t-1} \\ \mathbf{g}_j^T \mathbf{z}'_{t-1} + \mathbf{z}_j'' - 1 \geq 0, j=(N_{t-2}+1), \dots, N_{t-1} \\ \mathbf{z}_{t-1}'' \geq 0}} \left[ (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}_{t-1}'' \right] \\ \dot{\mathbf{z}}_{t-1} &= \begin{bmatrix} \dot{\mathbf{z}}'_{t-1} \\ \mathbf{z}''_{t-1} \end{bmatrix} \\ \hat{\mathbf{z}}_t &= \arg \min_{\mathbf{z}_t} \left[ (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \right] \end{aligned} \quad (1)$$

Задача определения переменной  $\dot{\mathbf{z}}_{t-1}$  в (1) является задачей общего квадратичного программирования небольшой размерности  $(n+1+(N_{t-1}-N_{t-2})) \times (n+1+(N_{t-1}-N_{t-2}))$  с ограничениями, решить которые можно довольно легко из-за небольшой размерности с помощью метода внутренней точки или симплекс метода. После нахождения переменной  $\dot{\mathbf{z}}_{t-1}$  в (1) оставшаяся переменная  $\hat{\mathbf{z}}_t$  определяется как минимум квадратичной формы  $(\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})$ , т.е.

$$\hat{\mathbf{z}}_t = \arg \min_{\mathbf{z}_t} \left[ (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \right] = \mathbf{A}_t \mathbf{z}'_{t-1}$$

Параметр  $\hat{c}_t$  есть не что иное, как значение минимума

$$\hat{c}_t = \tilde{c}_{t-1} + C\mathbf{e}_{t-1}^T \dot{\mathbf{z}}_{t-1}'' + (\dot{\mathbf{z}}_{t-1}' - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\dot{\mathbf{z}}_{t-1}' - \tilde{\mathbf{z}}_{t-1})$$

Остается выбрать матрицу  $\hat{\mathbf{Q}}_t$ , определяющую скорость квадратичного возрастания значения функции  $F_t(\mathbf{z}_t')$  при отклонения от точки минимума  $\hat{\mathbf{z}}_t$  в разных направлениях в пространстве векторного аргумента  $\mathbf{z}_t' \in \mathbb{R}^{n+1}$ .

$$F_t(\mathbf{z}_t') = \min_{\substack{\mathbf{z}_{t-1}', \mathbf{z}_{t-1}'' \\ \mathbf{g}_j^T \mathbf{z}_{t-1}' + z_j'' - 1 \geq 0, j = (N_{t-2} + 1), \dots, N_{t-1} \\ \mathbf{z}_{t-1}'' \geq 0}} \left[ \tilde{c}_{t-1} + (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1}) + C\mathbf{e}_{t-1}^T \mathbf{z}_{t-1}'' + \right. \\ \left. + (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}')^T \mathbf{U}_t (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}') \right] \quad (2)$$

Варьирование  $\mathbf{z}_t'$  будет, вообще говоря, приводить к изменению множеств активных ограничений, однако при малых отклонениях состав этих ограничений будет, скорее всего, оставаться тем же, что и в точках минимума. Тогда при малых отклонениях функция  $\hat{F}_t(\mathbf{z}_t')$  будет определяться как результат минимизации при тех ограничениях типа равенств, которые были активны в точке минимума как ограничения типа неравенств.

$$F_t(\mathbf{z}_t') = \min_{\mathbf{z}_{t-1}', \mathbf{z}_{t-1}''} \left[ \tilde{c}_{t-1} + (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1}) + C\mathbf{e}_{t-1}^T \mathbf{z}_{t-1}'' + (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}')^T \mathbf{U}_t (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}') \right] \\ \left\{ \begin{array}{l} \mathbf{g}_j^T \mathbf{z}_{t-1}' - 1 = 0, j = (N_{t-2} + 1), \dots, k_1 \\ z_j'' = 0, j = (N_{t-2} + 1), \dots, k_2 \\ \mathbf{g}_j^T \mathbf{z}_{t-1}' + z_j'' - 1 = 0, j = (k_2 + 1), \dots, N_{t-1} \end{array} \right.$$

$$\left\{ \begin{array}{l} \mathbf{G}'\mathbf{z}'_{t-1} - \mathbf{e}' = \mathbf{0}, \mathbf{G}' = [\mathbf{g}'_j^T]_{j=N_{t-2}+1}^{k_1}, \mathbf{e}' = [1]_{j=N_{t-2}+1}^{k_1} \\ \\ \mathbf{V}'\mathbf{z}''_{t-1} = \mathbf{0}, \mathbf{V}'((k_2 - N_{t-2}) \times (N_{t-1} - N_{t-2})) = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 & \dots & 0 \end{bmatrix} \\ \\ \mathbf{G}''\mathbf{z}'_{t-1} + \mathbf{V}''\mathbf{z}''_{t-1} - \mathbf{e}'' = \mathbf{0}, \mathbf{G}'' = [\mathbf{g}''_j^T]_{j=k_2+1}^{N_{t-1}}, \mathbf{V}''((N_{t-1} - k_2) \times (N_{t-1} - N_{t-2})) = \begin{bmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \\ \\ \mathbf{e}'' = [1]_{j=k_2+1}^{N_{t-1}} \end{array} \right.$$

$$L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1}) = \frac{1}{2} \tilde{c}_{t-1} + \frac{1}{2} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \tilde{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \frac{1}{2} C \mathbf{e}''^T \mathbf{V}'' \mathbf{z}''_{t-1} + \\ + \frac{1}{2} (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) - \boldsymbol{\mu}_{t-1}^T (\mathbf{G}' \mathbf{z}'_{t-1} - \mathbf{e}') - \mathbf{v}_{t-1}^T (\mathbf{G}'' \mathbf{z}'_{t-1} + \mathbf{V}'' \mathbf{z}''_{t-1} - \mathbf{e}'') - \boldsymbol{\eta}_{t-1}^T \mathbf{V}' \mathbf{z}''_{t-1},$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \mathbf{z}'_{t-1}} = \bar{\mathbf{Q}}_{t-1} \mathbf{z}'_{t-1} - \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} - \mathbf{A}_t^T \mathbf{U}_t \mathbf{z}'_t + \mathbf{A}_t^T \mathbf{U}_t \mathbf{A}_t \mathbf{z}'_{t-1} - \mathbf{G}'^T \boldsymbol{\mu}_{t-1} - \mathbf{G}''^T \mathbf{v}_{t-1} = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \mathbf{z}''_{t-1}} = \frac{1}{2} C \mathbf{V}''^T \mathbf{e}'' - \mathbf{V}''^T \mathbf{v}_{t-1} - \mathbf{V}'^T \boldsymbol{\eta}_{t-1} = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \boldsymbol{\mu}_{t-1}} = -\mathbf{G}' \mathbf{z}'_{t-1} + \mathbf{e}' = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \mathbf{v}_{t-1}} = -\mathbf{G}'' \mathbf{z}'_{t-1} - \mathbf{V}'' \mathbf{z}''_{t-1} + \mathbf{e}'' = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \boldsymbol{\eta}_{t-1}} = -\mathbf{V}' \mathbf{z}''_{t-1} = 0$$

Полученные условия приводят к системе линейных уравнений:

$$\left\{ \begin{array}{l} (\bar{\mathbf{Q}}_{t-1} + \mathbf{A}_t^T \mathbf{U}_t \mathbf{A}_t) \mathbf{z}'_{t-1} - \mathbf{G}'^T \boldsymbol{\mu}_{t-1} - \mathbf{G}''^T \mathbf{v}_{t-1} = \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \mathbf{A}_t^T \mathbf{U}_t \mathbf{z}'_t \\ \mathbf{G}' \mathbf{z}'_{t-1} = \mathbf{e}' \\ \mathbf{V}''^T \mathbf{v}_{t-1} + \mathbf{V}'^T \boldsymbol{\eta}_{t-1} = \frac{1}{2} C \mathbf{V}''^T \mathbf{e}'' \\ \mathbf{V}' \mathbf{z}''_{t-1} = 0 \\ \mathbf{G}'' \mathbf{z}'_{t-1} + \mathbf{V}'' \mathbf{z}''_{t-1} = \mathbf{e}'' \end{array} \right.$$

Или в обобщенно матричной форме:





$$\bar{\mathbf{Q}}_t = \hat{\mathbf{Q}}_t + \mathbf{Q}_t^0, \tilde{\mathbf{z}}_t = (\bar{\mathbf{Q}}_t)^{-1} (\mathbf{Q}_t^0 \mathbf{z}_t^0 + \hat{\mathbf{Q}}_t \hat{\mathbf{z}}_t), \tilde{c}_t = c'_t$$

квадратичных аналогов функций Беллмана  $\tilde{J}_j(\mathbf{x}_j)$

$$\tilde{J}_t(\mathbf{z}_t) \cong \zeta_t(\mathbf{z}'_t) + \chi(\mathbf{z}''_t) + \hat{F}_t(\mathbf{z}'_t) = \tilde{c}_t + (\mathbf{z}'_t - \tilde{\mathbf{z}}_t)^T \bar{\mathbf{Q}}_t (\mathbf{z}'_t - \tilde{\mathbf{z}}_t) + \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t$$

в направлении  $t = 0, \dots, T$ , начиная с исходных присвоений  $\tilde{\mathbf{Q}}_0 = \mathbf{Q}_0^0, \tilde{\mathbf{z}}_0 = \mathbf{z}_0^0, \tilde{c}_0 = 0$ , с использованием формулы (17), связанной с решением на каждом шаге соответствующей задачи квадратичного программирования (20) размерности  $(n+2)$ , и далее формул (18), (19).

Оптимальное значение векторной переменной в последней вершине  $t = T$  получается непосредственно с помощью минимизирования квадратичной функции Беллмана в последней вершине при соответствующих ограничениях

$$\bar{\mathbf{z}}_T = \arg \min_{\substack{\mathbf{z}_T \\ \mathbf{g}_j^T \mathbf{z}'_T + z_j'' - 1 \geq 0, j = (N_{T-1}+1), \dots, N_T \\ \mathbf{z}''_T \geq 0}} \tilde{J}_T(\mathbf{x}_T) = \arg \min_{\substack{\mathbf{z}_T \\ \mathbf{g}_j^T \mathbf{z}'_T + z_j'' - 1 \geq 0, j = (N_{T-1}+1), \dots, N_T \\ \mathbf{z}''_T \geq 0}} \left[ \tilde{c}_T + (\mathbf{z}'_T - \tilde{\mathbf{z}}_T)^T \bar{\mathbf{Q}}_T (\mathbf{z}'_T - \tilde{\mathbf{z}}_T) + \mathbf{C} \mathbf{e}_T^T \mathbf{z}''_T \right]$$

Таким образом, мы получим решение задачи фильтрации  $\hat{\mathbf{z}}_T$ .

Для нахождения задачи интерполяции, в направлении  $t = T, \dots, 0$  вычисляются оптимальные значения переменных по обратному рекуррентному соотношению

$$\bar{\mathbf{z}}_t = (\mathbf{z}_{t-1}(\bar{\mathbf{z}}_t)) = \arg \min_{\substack{\mathbf{z}_{t-1} \\ \mathbf{z}_{t-1} \in \mathbf{Z}_{t-1}}} \left[ \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + \bar{J}_{t-1}(\mathbf{z}_{t-1}) \right] = \arg \min_{\substack{\mathbf{z}_{t-1} \\ \mathbf{z}_{t-1} \in \mathbf{Z}_{t-1}}} \left[ (\mathbf{z}'_t - \hat{\mathbf{z}}_t)^T \hat{\mathbf{Q}}_t (\mathbf{z}'_t - \hat{\mathbf{z}}_t) + \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t \right]$$

где параметры  $\hat{\mathbf{Q}}_t$ ,  $\hat{\mathbf{z}}_t$  определяются из соотношений (17)–(19).

Достоинством указанного алгоритма является то, что для построения решающего правила при поступлении новой порции объектов необходимо лишь знание параметров классификатора в данный момент и не требуется хранение обучающего множества. Данный алгоритм является достаточно точным, что было подтверждено экспериментами на реальных данных. Скорость его работы также вполне приемлема для задач распознавания, в которых данные поступают группами объемом до нескольких сотен.

Недостаток метода состоит в том, что возможность численной реализации процедуры динамического программирования основана на

предположении о существовании параметрического семейства, которому на каждом шаге процедуры принадлежат функции Беллмана. В данной задаче такого параметрического семейства не существует, и приходится аппроксимировать неквадратичные функции Беллмана их квадратичными аналогами, что неизбежно приводит к ухудшению качества распознавания. Однако, на больших массивах данных из-за необходимости решения для каждого отсчета оптимизационной задачи (20) время поиска может оказаться неудовлетворительным.

## 5. Обобщение модели логистической регрессии на случай нестационарной генеральной совокупности

### 5.1. Модель логистической регрессии в задаче обучения распознаванию образов

Одним из методов восстановления эмпирической зависимости  $y(\omega): \Omega \rightarrow Y$  является логистическая регрессия, которая, несмотря на присутствие в названии слова «регрессия», предназначена для решения задач классификации.

Логистическая регрессия основана на следующей модели, описывающей апостериорные вероятности принадлежности объектов классам[24]:

$$\begin{aligned} P(y=1|\mathbf{x}) &= \sigma(z) \\ P(y=-1|\mathbf{x}) &= 1 - \sigma(z), \end{aligned} \quad (21)$$

где  $\sigma(z) = \frac{1}{1+e^{-z}}$  – логистическая функция (Рисунок 4),  $z = x_1\theta_1 + x_2\theta_2 + \dots + x_n\theta_n$  – линейная модель регрессии. Для вышеописанной задачи двуклассового распознавания с учетом(2) выражения(21) приобретают вид:

$$\begin{aligned} P(y=1|\mathbf{x}) &= \sigma(-(\mathbf{a}^T \mathbf{x} + b)) \\ P(y=-1|\mathbf{x}) &= \sigma(\mathbf{a}^T \mathbf{x} + b) \end{aligned} \quad (22)$$

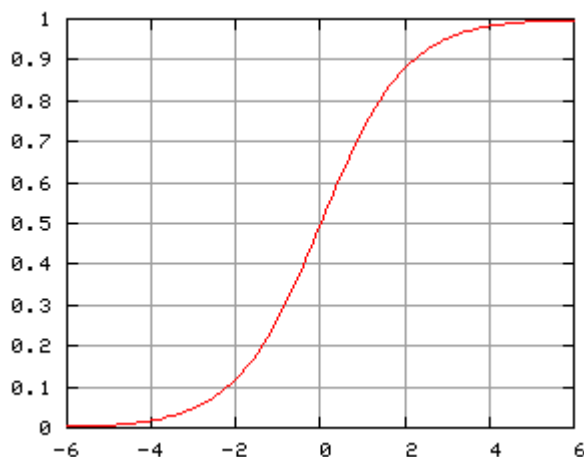


Рисунок 4 - Логистическая функция

Для оценки оптимальных значений параметров  $\mathbf{a}$  и  $b$  обычно используется метод максимального правдоподобия, в соответствии с

которым для их нахождения требуется оптимизировать следующий критерий:

$$(\hat{\mathbf{a}}, \hat{b}) = \arg \max_{\mathbf{a}, b} \prod_{j=1}^N P(y = y_j | \mathbf{x} = \mathbf{x}_j),$$

что эквивалентно:

$$(\hat{\mathbf{a}}, \hat{b}) = \arg \max_{\mathbf{a}, b} \sum_{j=1}^N \log P(y = y_j | \mathbf{x} = \mathbf{x}_j). \quad (23)$$

Решение задачи (23) возможно только итерационными методами, такими как градиентный спуск или метод Ньютона, что является недостатком логистической модели. Однако, после определения параметров решающего правила появляется возможность оценивать вероятности принадлежности объектов к каждому из двух классов.

Улучшить обобщающую способность логистической регрессии позволяет применение регуляризации. Идея ее в том, что направляющий вектор гиперплоскости  $\mathbf{a}$  и параметр положения  $b$  полагаются случайными с некоторой априорной плотностью распределения  $p(\mathbf{a}, b)$ , тогда применение метода максимизации апостериорной вероятности приводит к следующему оптимизационному критерию:

$$(\hat{\mathbf{a}}, \hat{b}) = \arg \max_{\mathbf{a}, b} \prod_{j=1}^N P(y = y_j | \mathbf{x} = \mathbf{x}_j, \mathbf{a}, b) p(\mathbf{a}, b)$$

## 5.2. Вероятностное обоснование модели логистической регрессии

Запишем логистические функции  $P(y=1|\mathbf{x})$  и  $P(y=-1|\mathbf{x})$  (22), описывающие апостериорные вероятности принадлежности объекта к каждому из двух классов, в следующем виде:

$$\begin{aligned} \phi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) &= P(y=1|\mathbf{x}) = \frac{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2)}{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)} \\ \phi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2) &= P(y=-1|\mathbf{x}) = \frac{\varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)}{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)}, \end{aligned} \quad (24)$$

где

$$\begin{aligned}\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) &\propto \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right) \\ \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2) &\propto \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b + 1)^2\right).\end{aligned}\quad (25)$$

Нетрудно убедиться, что функции (24) полностью эквивалентны (22). В самом деле, например, для апостериорной вероятности распределения класса  $y = 1$

$$\begin{aligned}\phi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) &= \frac{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2)}{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)} = \\ &= \frac{\exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right)}{\exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b + 1)^2\right) + \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right)} = \\ &= \frac{1}{1 + \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b + 1)^2 + \frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right)} = \\ &= \frac{1}{1 + \exp\left(-\frac{1}{\sigma^2}(\mathbf{a}^T \mathbf{x} + b)\right)} = \frac{1}{1 + \exp\left(-\frac{2}{\sigma^2}(\mathbf{a}^T \mathbf{x} + b)\right)}\end{aligned}$$

Аналогично для логистической функции противоположного класса  $y = -1$ .

Тогда совместная условная апостериорная вероятность для всей обучающей совокупности  $\{\mathbf{X}, \mathbf{Y}\}$ :

$$\begin{aligned}\Phi(Y | X, \mathbf{a}, b, \sigma^2) &= \prod_{j: y_j=1} \phi_1(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) \prod_{j: y_j=-1} \phi_{-1}(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) = \\ &= \frac{\prod_{j: y_j=1} \varphi_1(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) \prod_{j: y_j=-1} \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b, \sigma^2)}{\prod_j (\varphi_1(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b, \sigma^2))}\end{aligned}$$

Внесем значение индекса класса объекта в квазинормальные распределения (25) и перепишем функции **Ошибка! Закладка не определена.** в более краткой форме:

$$\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) = \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(1 | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(-1 | \mathbf{x}, \mathbf{a}, b, \sigma^2)}, \quad (26)$$

где

$$\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) \propto \exp\left[-\frac{1}{2\sigma^2}\left(1 - y(\mathbf{a}^T \mathbf{x} + b)\right)^2\right] \quad (27)$$

Критерий обучения, построенный по принципу максимизации апостериорной вероятности,

$$\Phi(Y | X, \mathbf{a}, b, \sigma^2) \rightarrow \max_{\mathbf{a}, b},$$

$$\Phi(Y | X, \mathbf{a}, b, \sigma^2) = \prod_{j=1}^N \phi(y_j | \mathbf{x}_j, \mathbf{a}, b, \sigma^2) = \prod_{j=1}^N \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}$$

Максимизация функции  $\Phi(Y | X, \mathbf{a}, b, \sigma^2)$  эквивалентна максимизации ее логарифма

$$\log \Phi(Y | X, \mathbf{a}, b, \sigma^2) \rightarrow \max_{\mathbf{a}, b},$$

$$\begin{aligned} \log \Phi(Y | X, \mathbf{a}, b, \sigma^2) &= \log \prod_{j=1}^N \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)} = \\ &= \sum_{j=1}^N \log \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)} \end{aligned}$$

С учетом Ошибка! Закладка не определена. Ошибка! Закладка не определена.

$$\begin{aligned} G(\mathbf{a}, b | \sigma, (y_i, \mathbf{x}_i)_{i=1}^N) &= \sum_{i=1}^N \left( \frac{1}{2\sigma^2} (1 - y_i(\mathbf{a}^T \mathbf{x}_i + b))^2 \right) + \\ &+ \sum_{i=1}^N \log \left[ \exp \left( -\frac{1}{2\sigma^2} (1 + \mathbf{a}^T \mathbf{x}_i + b)^2 \right) + \exp \left( -\frac{1}{2\sigma^2} (1 - \mathbf{a}^T \mathbf{x}_i - b)^2 \right) \right] \rightarrow \min_{\mathbf{a}, b} \end{aligned} \quad (28)$$

Можно доказать, что для критерия (28) имеет место

**Теорема 3.** Если априорные плотности распределения классов имеют вид (26) – (27), то при  $N \rightarrow \infty$  и равнонаполненности классов решение оптимизационной задачи (27) сводится к оптимизации критерия:

$$\hat{G}(\mathbf{a}, b | \sigma, (y_i, \mathbf{x}_i)_{i=1}^N) = \sum_{i=1}^N \left( \frac{1}{2\sigma^2} (1 - y_i(\mathbf{a}^T \mathbf{x}_i + b))^2 \right) \rightarrow \min_{\mathbf{a}, b}.$$

### 5.3. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода логистической регрессии

В терминах логистической модели представление о нестационарной генеральной совокупности, изложенное в п. 2.2 выражается в виде

предположения, что параметры логистических функций (26), описывающих апостериорные вероятности принадлежности объекта к каждому из двух классов, изменяются во времени:

$$\phi(y | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2) = \frac{\varphi(y | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2)}{\varphi(1 | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2) + \varphi(-1 | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2)}, \quad (29)$$

где

$$\varphi(y | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2) \propto \exp \left[ -\frac{1}{2\sigma^2} \left( 1 - y(\mathbf{a}_t^T \mathbf{x} + b_t) \right)^2 \right]. \quad (30)$$

Совместная условная апостериорная вероятность принадлежности объектов в составе обучающей совокупности выражается как произведение:

$$\Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2) = \prod_{j=1}^{N_t} \phi(y_j | \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2)$$

где согласно(29)

$$\phi(y_j | \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) = \frac{\exp \left[ -\frac{1}{2\sigma^2} \left( 1 - y_j(\mathbf{a}_t^T \mathbf{x}_j + b_t) \right)^2 \right]}{\exp \left[ -\frac{1}{2\sigma^2} \left( 1 - (\mathbf{a}_t^T \mathbf{x}_j + b_t) \right)^2 \right] + \exp \left[ -\frac{1}{2\sigma^2} \left( 1 + (\mathbf{a}_t^T \mathbf{x}_j + b_t) \right)^2 \right]},$$

Совместная априорная плотность распределения параметров за весь период времени определяется как произведение плотностей для отдельных отсчетов (5):

$$\Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) = \prod_{t=1}^T \psi_t(\mathbf{a}_t, b_t | \mathbf{a}_{t-1}, b_{t-1})$$

Апостериорное распределение параметров разделяющих гиперплоскостей после наблюдения обучающей совокупности определяется формулой Байеса:

$$\begin{aligned} P(\mathbf{a}_t, b_t, t = 0, \dots, T | \mathbf{X}, \mathbf{Y}) &= \frac{\Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2, t = 1, \dots, T)}{\int \dots \int \dots \int \Psi(\mathbf{a}'_t, b'_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}, \mathbf{a}'_t, b'_t, t = 1, \dots, T) db_0 \dots db_T d\mathbf{a}_0 \dots d\mathbf{a}_T} = \\ &= \frac{\Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2, t = 1, \dots, T)}{F(\mathbf{Y} | \mathbf{X})} \propto \Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2, t = 1, \dots, T) \end{aligned}$$

А обучение будем понимать как вычисление байесовской оценки параметров разделяющих гиперплоскостей:

$$\begin{aligned}
& (\mathbf{a}_t, b_t, t=1, \dots, T \mid \mathbf{X}, \mathbf{Y}, \sigma^2, d, d') = \arg \max \Phi(\mathbf{Y} \mid \mathbf{X}, \mathbf{a}_t, b_t, \sigma^2, t=1, \dots, T) \Psi(\mathbf{a}_t, b_t, t=1, \dots, T) = \\
& = \arg \max \left[ \ln \Phi(\mathbf{Y} \mid \mathbf{X}, \mathbf{a}_t, b_t, \sigma^2, t=1, \dots, T) + \ln \Psi(\mathbf{a}_t, b_t, t=1, \dots, T) \right] = \\
& = \arg \max \left[ \sum_{t=1}^T \sum_{j=N_{t-1}+1}^{N_t} \ln \varphi(y_j \mid \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) - \sum_{i=1}^N \ln \left[ \varphi(1 \mid \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) + \varphi(-1 \mid \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) \right] + \right. \\
& \quad \left. + \sum_{t=1}^T \ln \psi_t(\mathbf{a}_t, b_t \mid \mathbf{a}_{t-1}, b_{t-1}) \right]
\end{aligned}$$

Используя Теорема 3, получим следующую эквивалентную оптимизационную задачу:

$$\begin{aligned}
J(\mathbf{a}_t, b_t, t=1, \dots, T) = & \left[ \frac{1}{2\sigma^2} \sum_{t=1}^T \sum_{j=N_{t-1}+1}^{N_t} (1 - y_j(\mathbf{a}_t^T \mathbf{x}_j + b_t))^2 + \right. \\
& \left. + \frac{1}{d} \sum_{t=2}^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1})^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1}) + \frac{1}{2d'} \sum_{t=2}^T (b_t - b_{t-1})^2 \right] \rightarrow \min_{(\mathbf{a}_t, b_t)_{t=1}^T}
\end{aligned}$$

Именно этот критерий мы будем в дальнейшем использовать, однако удобнее записать его в более общем виде:

$$J(\mathbf{z}_1, \dots, \mathbf{z}_T) = \sum_{t=1}^T (\mathbf{z}_t - \mathbf{z}_t^0)^T \mathbf{Q}_t (\mathbf{z}_t - \mathbf{z}_t^0) + \sum_{t=2}^T (\mathbf{z}_t - \mathbf{A} \mathbf{z}_{t-1})^T \mathbf{U} (\mathbf{z}_t - \mathbf{A} \mathbf{z}_{t-1}) \rightarrow \min_{\mathbf{z}_1, \dots, \mathbf{z}_T}, \quad (31)$$

где введены следующие обозначения:

$$\begin{aligned}
\mathbf{z}_t = & \begin{bmatrix} \mathbf{a}_t \\ b_t \end{bmatrix}; \quad \mathbf{z}_t^0 = (\mathbf{Q}_t^T \mathbf{Q}_t)^{-1} \mathbf{Q}_t^T \sum_{j=N_{t-1}}^{N_t} \mathbf{g}_j; \quad \mathbf{Q}_t = C \sum_{j=N_{t-1}}^{N_t} \mathbf{g}_j \mathbf{g}_j^T; \quad \mathbf{g}_j = \begin{bmatrix} y_j \mathbf{x}_j \\ y_j \end{bmatrix}, \quad t=1, \dots, T \\
\mathbf{U} = & \begin{bmatrix} \frac{1}{d} & 0 & \dots & 0 & 0 \\ 0 & \frac{1}{d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{d} & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{d'} \end{bmatrix}; \quad \mathbf{A} = \begin{bmatrix} \sqrt{1-d} & 0 & \dots & 0 & 0 \\ 0 & \sqrt{1-d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \sqrt{1-d} & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}.
\end{aligned}$$

#### 5.4. Численная реализация процедуры динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью модели логистической регрессии

Парно-сепарабельный критерий (31) имеет вид

$$J(\mathbf{z}_1, \dots, \mathbf{z}_T) = \sum_{t=1}^T \zeta_t(\mathbf{z}_t) + \sum_{t=2}^T \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) \rightarrow \min, \quad \mathbf{z}_t \in Z_t, \quad (32)$$



где области допустимых значений переменных определяются условиями

$$Z_t = \{ \mathbf{z}_t \in \mathbb{R}^{n+1}, t = 1, \dots, T \}$$

Функции  $\zeta_t(\mathbf{z}_t), \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t)$  являются квадратичными

$$\begin{aligned}\zeta_t(\mathbf{z}_t) &= (\mathbf{z}_t - \mathbf{z}_t^0)^T \mathbf{Q}_t (\mathbf{z}_t - \mathbf{z}_t^0) \\ \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) &= (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \mathbf{U} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})\end{aligned}$$

Рассмотрим способ решения задачи (32) на основе принципа динамического программирования [25], который хотя и адекватен парно-сепарабельным целевым функциям, но был создан для решения задач оптимизации с дискретными аргументами.

Применительно к задаче(32), центральная идея метода динамического программирования заключается в понятии последовательности функций Беллмана  $J_t(\mathbf{z}_t) = \min_{\mathbf{z}_1, \dots, \mathbf{z}_{t-1}} J_t(\mathbf{z}_1, \dots, \mathbf{z}_t), \mathbf{z}_s \in Z_s, s = 1, \dots, t-1$ , связанных с частичными критериями

$$J(\mathbf{z}_1, \dots, \mathbf{z}_t) = \sum_{s=1}^t \zeta_s(\mathbf{z}_s) + \sum_{s=2}^t \gamma_s(\mathbf{z}_{s-1}, \mathbf{z}_s),$$

имеющими такую же структуру, как и полная целевая функция(32), но определенными на множестве переменных  $Z_t = (\mathbf{z}_s, s = 1, \dots, t)$ . Нетрудно заметить, что

$$J_t(\mathbf{z}_1, \dots, \mathbf{z}_t) = \zeta_t(\mathbf{z}_t) + \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + J_{t-1}(\mathbf{z}_1, \dots, \mathbf{z}_{t-1})$$

При  $t=1$ , очевидно:

$$J_1(\mathbf{z}_1) = \zeta_1(\mathbf{z}_1), \quad (33)$$

а в последний момент времени  $t=T$  функция Беллмана определяется следующим выражением

$$J(\mathbf{z}_1, \dots, \mathbf{z}_T) = J_T(\mathbf{z}_1, \dots, \mathbf{z}_T) \quad (34)$$

Фундаментальное свойство функции Беллмана

$$\begin{aligned}\tilde{J}_t(\mathbf{z}_t) &= \zeta_t(\mathbf{z}_t) + \min_{\mathbf{z}_{t-1}} [\gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + \tilde{J}_{t-1}(\mathbf{z}_{t-1})], \\ t &= 2, \dots, T, \mathbf{z}_{t-1} \in Z_{t-1},\end{aligned} \quad (35)$$

будем называть прямым рекуррентным соотношением[25], а функцию

$$(\mathbf{z}_{t-1}) = (\mathbf{z}_{t-1}(\mathbf{z}_t)) = \arg \min_{\mathbf{z}_{t-1}} [\gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + \tilde{J}_{t-1}(\mathbf{z}_{t-1})], \mathbf{z}_{t-1} \in Z_{t-1} \quad (36)$$

будем называть обратным рекуррентным соотношением.

В основе процедуры оптимизации лежит предположение, что существуют, во-первых, достаточно эффективный способ решения частных задач оптимизации, входящих в (35)и, во-вторых, подходящая компактная форма представления функций Беллмана или обратных рекуррентных соотношений (36)позволяющая хранить эти функции в памяти.

Процедура оптимизации пробегает дважды по всем отсчетам вперед от начала к концу сигнала  $t = 1, \dots, T$  и затем назад от конца к началу  $t = T, \dots, 1$ .

Процедура начинается со значения первой функции Беллмана  $t = 1$ ,  $\tilde{J}_1(\mathbf{z}_1) = \zeta_1(\mathbf{z}_1)$  далее осуществляется пересчет функций Беллмана для последующих отсчетов  $t = 1, \dots, T$  в соответствии с прямым рекуррентным соотношением(35). При этом функции Беллмана или обратные рекуррентные соотношения (36)должны быть запомнены для всех  $t = 1, \dots, T$ . Функция Беллмана для последней переменной  $\tilde{J}_T(\mathbf{z}_T)$ , полученная на последнем шаге прямого хода алгоритма, непосредственно указывает ее оптимальное значение:

$$\hat{\mathbf{z}}_T = \arg \min_{\mathbf{z}_T} \hat{J}_T(\mathbf{z}_T), \mathbf{z}_T \in Z_T$$

Таким образом, в силу (34)мы получим, что найденное  $\hat{\mathbf{z}}_T$  является решением задачи фильтрации.

На обратном ходе, по мере того, как процедура последовательно проходит от конца сигнала к началу  $t = T, \dots, 2$ , уже найденное оптимальное значение векторной переменной в каждой очередной вершине позволяет, в свою очередь, определить оптимальное значение переменной в непосредственно предшествующей вершине. Последовательное вычисление оптимальных значений переменных обеспечивается обратными рекуррентными соотношениями, найденными и сохраненными в процессе

прямого хода алгоритма либо непосредственно в виде (36) либо косвенно определяемыми функциями Беллмана(35), запоминаемыми вместо них:

$$\hat{\mathbf{z}}_{t-1} = \tilde{\mathbf{z}}_{t-1}(\hat{\mathbf{z}}_t)$$

Найденные оптимальные значения  $\hat{\mathbf{z}}_1, \dots, \hat{\mathbf{z}}_T$  являются решением задачи интерполяции.

Поскольку в оптимизационной задаче(31)все составляющие парно-сепарабельной целевой функции являются квадратичными, то и функции Беллмана также являются квадратичными, их параметры можно легко пересчитывать, и, соответственно, может быть реализована процедура динамического программирования.

Следовательно, функции Беллмана будут определяться следующим выражением

$$\tilde{J}_t(\mathbf{z}_t) = (\mathbf{z}_t - \tilde{\mathbf{z}}_t)^T \bar{\mathbf{Q}}_t (\mathbf{z}_t - \tilde{\mathbf{z}}_t) + \tilde{c}_t, \quad (37)$$

параметры которого для момента  $t=1$  будут принимать тривиальные значения:

$$\tilde{\mathbf{z}}_1 = \mathbf{z}_1^0; \bar{\mathbf{Q}}_1 = \mathbf{Q}; \tilde{c}_1 = 0.$$

Пусть

$$F_t(\mathbf{z}_t) = \min_{\mathbf{z}_{t-1}} \left[ (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \mathbf{U} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1}) + (\mathbf{z}_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \tilde{c}_{t-1} \right],$$

соответственно

$$\frac{\partial F_t(\mathbf{z}_t)}{\partial \mathbf{z}_{t-1}} = -2\mathbf{A}\mathbf{U}\mathbf{z}_t + 2\mathbf{A}\mathbf{U}\mathbf{A}\mathbf{z}_{t-1} + 2\bar{\mathbf{Q}}_{t-1}\mathbf{z}_{t-1} - 2\bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1} = (\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}).$$

Согласно необходимому условию экстремума получаем:

$$\begin{aligned} F_t(\mathbf{z}_t) = & \left[ \mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right]^T \mathbf{U} \left[ \mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right] + \\ & + \left[ (\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) - \tilde{\mathbf{z}}_{t-1} \right]^T \bar{\mathbf{Q}}_{t-1} \left[ (\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) - \tilde{\mathbf{z}}_{t-1} \right] + \tilde{c}_{t-1} \end{aligned}$$

где выделим две составляющие:

$$\begin{aligned} & \left( \mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right)^T \mathbf{U} \left( \mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right) = \\ & = \left[ \mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right]^T \mathbf{U} \left[ \mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right] \quad (38) \end{aligned}$$

и

$$\begin{aligned} & \left( \mathbf{z}_t - \ddot{\mathbf{z}}_t \right)^T \ddot{\mathbf{Q}}_t \left( \mathbf{z}_t - \ddot{\mathbf{z}}_t \right) = \\ & = \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \left( \mathbf{AUz}_t + \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right) - \tilde{\mathbf{z}}_{t-1} \right]^T \bar{\mathbf{Q}}_{t-1} \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \left( \mathbf{AUz}_t + \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right) - \tilde{\mathbf{z}}_{t-1} \right] \end{aligned} \quad (39)$$

Преобразуем **Ошибка! Закладка не определена.**

$$\begin{aligned} & \left( \mathbf{z}_t - \dot{\mathbf{z}}_t \right)^T \dot{\mathbf{Q}}_t \left( \mathbf{z}_t - \dot{\mathbf{z}}_t \right) = \\ & = \left[ \left( \mathbf{I} - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right) \mathbf{z}_t - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right]^T \mathbf{U} \cdot \\ & \quad \cdot \left[ \left( \mathbf{I} - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right) \mathbf{z}_t - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right] \end{aligned}$$

откуда:

$$\begin{aligned} \dot{\mathbf{Q}}_t &= \left[ \mathbf{I} - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right]^T \mathbf{U} \left[ \mathbf{I} - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right] \\ \dot{\mathbf{z}}_t &= \dot{\mathbf{Q}}_t^{-1} \cdot \left( \mathbf{I} - \mathbf{A} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right)^T \mathbf{UA} \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1}. \end{aligned}$$

Из **Ошибка! Закладка не определена.** следует:

$$\begin{aligned} & \left( \mathbf{z}_t - \ddot{\mathbf{z}}_t \right)^T \ddot{\mathbf{Q}}_t \left( \mathbf{z}_t - \ddot{\mathbf{z}}_t \right) = \\ & = \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AUz}_t - \left( \mathbf{I} - \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} \right]^T \bar{\mathbf{Q}}_{t-1} \cdot \\ & \quad \cdot \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AUz}_t - \left( \mathbf{I} - \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} \right] \end{aligned}$$

откуда в свою очередь

$$\begin{aligned} \ddot{\mathbf{Q}}_t &= \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right] \\ \ddot{\mathbf{z}}_t &= \ddot{\mathbf{Q}}_t^{-1} \cdot \left[ \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left( \mathbf{I} - \left( \mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} \end{aligned}$$

Подставляя **Ошибка! Закладка не определена.** с учетом **Ошибка!**

**Закладка не определена.** определяем параметры функций Беллмана (37) на каждом шаге

$$\begin{aligned}
\bar{\mathbf{Q}}_t &= \mathbf{Q}_t + \dot{\mathbf{Q}}_t + \ddot{\mathbf{Q}}_t = \\
&= \left[ \mathbf{I} - \mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right]^T \mathbf{U} \left[ \mathbf{I} - \mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right] + \\
&\quad + \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right] + \mathbf{Q}_t = \\
&= \mathbf{U} - 2\mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AUU} + \mathbf{UA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AUA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} + \\
&\quad + \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right] + \mathbf{Q}_t = \\
&= \mathbf{U} - 2\mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AUU} + \mathbf{UA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1}) (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} + \mathbf{Q}_t = \\
&= \mathbf{U} - \mathbf{AUA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{U} + \mathbf{Q}_t = \bar{\mathbf{Q}}_{t-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{U} + \mathbf{Q}_t
\end{aligned}$$

и

$$\begin{aligned}
\tilde{\mathbf{z}}_t &= \bar{\mathbf{Q}}_t^{-1} \left( \mathbf{Q}_t \mathbf{z}_t^0 + \dot{\mathbf{Q}}_t \tilde{\mathbf{z}}_t + \ddot{\mathbf{Q}}_t \tilde{\mathbf{z}}_t \right) = \bar{\mathbf{Q}}_t^{-1} \left[ \left( \mathbf{I} - \mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right)^T \mathbf{UA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \right. \\
&\quad \left. + \left( (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right)^T \bar{\mathbf{Q}}_{t-1} \left( \mathbf{I} - (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} + \mathbf{Q}_t \mathbf{z}_t^0 \right] = \\
&= \bar{\mathbf{Q}}_t^{-1} \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \cdot \left( 2\mathbf{I} - (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1}) \right) + \mathbf{Q}_t \mathbf{z}_t^0 \right] = \\
&= \bar{\mathbf{Q}}_t^{-1} \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \mathbf{Q}_t \mathbf{z}_t^0 \right]
\end{aligned}$$

Таким образом, окончательно

$$\begin{aligned}
\bar{\mathbf{Q}}_t &= \bar{\mathbf{Q}}_{t-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{U} + \mathbf{Q}_t \\
\tilde{\mathbf{z}}_t &= \bar{\mathbf{Q}}_t^{-1} \left[ (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \mathbf{Q}_t \mathbf{z}_t^0 \right]
\end{aligned}$$

Значение целевой переменной в последний момент времени  $\tilde{\mathbf{z}}_T$  непосредственно представляет решение задачи фильтрации. Задача интерполяции решается путем рекуррентного пересчета при условии сохранения в памяти значений  $\bar{\mathbf{Q}}_{t-1}$ ,  $t = T, \dots, 1$

$$\hat{\mathbf{z}}_{t-1} = (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{AU} \hat{\mathbf{z}}_t + \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1})$$

## 6. Экспериментальное исследование

### 6.1. Экспериментальное исследование на модельных данных

Для исследования работы метода нами было создано искусственное множество данных, образованное двумя нормальными распределениями. Для получения свойства нестационарности центры этих распределений поворачиваются относительно начала координат. В начальный момент времени дисперсии распределений одинаковы и равны 1, математические ожидания – 3.5 и 6.5, соответственно. Создаваемые объекты описываются двумя признаками и индексом класса  $\{-1;1\}$ . Обучающее множество составили признаковые описания объектов 50-и моментов времени, по 100 объектов в каждом. Контрольная выборка содержит 2000 объектов, соответствующих 51-у отсчету. В Таблице 1 содержится процент ошибочной классификации объектов каждого класса.

Таблица 1 – Экспериментальные результаты:

модельные данные

Значения параметров $d$ и $d'$	Ошибка классификации объектов класса -1, %	Ошибка классификации объектов класса +1, %
$d \rightarrow 0; d' \rightarrow 0$	$\rightarrow 0$	$\rightarrow 100$
$d=1; d'=1$	1.0092	3.1257
$d=10^{-8}; d'=10^{-8}$	1,0801	2,846

Первые две строки таблицы описывают экстремальные случаи:

- $d \rightarrow 0; d' \rightarrow 0$  – большое значение штрафа в критерии не позволяет адаптироваться к происходящим в генеральной совокупности изменениям;
- $d=1; d'=1$  – модель сразу «забывает» информацию, содержащуюся в ранее полученных данных (поскольку данные соответствуют идеальному случаю для выбранной модели ошибка невелика).

## 6.2. Экспериментальное исследование на реальных данных

Для следующего эксперимента мы выбрали коллекцию данных, собранных с почтовых фильтров, и описывающих электронные письма[26].

В этом множестве содержатся записи о 4601 электронном сообщении, каждое из которых описывается 58 признаками. Значения признаков, характеризующих объекты-письма, являются непрерывными и показывают частоту встречаемости отдельных элементов (слов или символов) в тексте письма или длину непрерывной последовательности прописных букв. Кроме этого, о каждом письме известно, является ли оно рекламным («спам») или же нет, что описывается меткой 1 или 0 соответственно. Процентное соотношение объектов в классах: спам – 1813 (39.4%), не-спам – 2788 (60.6%).

Предварительно была проведена замена меток класса 0 на -1, данные были стандартизованы. Обучающая выборка составлена из 3600 объектов с сохранением исходного соотношения классов. Предполагалось, что каждый момент времени подавалось по 400 объектов. Ошибка классификации вычислялась как процент ошибочно классифицированных объектов к размеру контрольной выборки.

Полученные на контрольном множестве результаты сравнивались с результатами некоторых алгоритмов для распознавания при смещении решающего правила из программного пакета Massive Online Analysis (MOA) [27]:

- OzaBagASHT [19] – метод bagging с адаптивными деревьями Хевдинга; после достижения максимального размера дерево строится заново, начиная с корня. Была проведена серия экспериментов при различных значениях параметра, определяющего максимальное количество листьев в дереве, результаты представлены в Таблице 2.

Таблица 2 – Величина ошибки классификации  
при настройке OzaBagASHT

-s	6	8	10	12	14	16	18	20	22	24	26	28	30
Ошибка	39,46	30,57	30,1	31,77	28,87	30,87	25,77	30,17	30,17	22,28	31,47	32,47	31,17

- OzaBagAdwin [**Ошибка! Закладка не определена.**]- метод agging с обнаружением изменений в данных по алгоритму ADWIN [13]; в качестве классификатора были выбраны решающие деревья с адаптивным байесовским правилом. Была проведена серия экспериментов при различных значениях параметра, определяющего количество классификаторов в наборе, результаты представлены в Таблице 3.

Таблица 3 – Величина ошибки классификации  
при настройке OzaBagAdwin

-s	6	8	10	12	14	16	18	20	22	24	26
Ошибка	22,278	23,278	22,877	23,776	26,773	22,977	20,879	22,078	22,677	22,278	24,177

- SingleClassifierDrift[ 28 ] – одиночный классификатор; использовалось обнаружение изменений по методу EDDM, в качестве классификатора было выбрано решающее дерево с адаптивным байесовским правилом.
- AdaHoeffdingOptionTree – адаптивное дерево Хевдинга с дополнительными (option)узлами; максимальное количество таких узлов – 50, в ходе испытаний не выявлено влияния этого параметра на величину ошибки.
- LimAttClassifier [20] – ансамбль ограниченных деревьев Хевдинга, каждое из которых строится на своем наборе признаков;



результат получается объединением предсказанных каждым деревом вероятностей классов с использованием сигмовидного персептрона. При обнаружении изменений использовался подход bagging (позволило увеличить точность классификации). Количество признаков в наборе для каждого классификатора равно 2 (при 4 признаках в наборе результат не изменялся).

- Для определения значений параметров представленного нами метода было проведено несколько пробных экспериментов, в результате:  $d=10^{-8}$ ;  $d'=10^{-8}$ ,  $C=1$ .

В таблице 4 приведены лучшие из полученных результатов классификации для каждого метода.

Таблица 4 – Экспериментальные данные:  
база данных электронных писем

Алгоритм	Ошибка классификации, %
OzaBagASHT	22,278
OzaBagAdwin	20,879
SingleClassifierDrift	39.361
AdaHoeffdingOptionTree	23.876
LimAttClassifier	29,271
Предложенный метод	14,785

Как видно, на реальных данных разработанный алгоритм показал хорошие результаты распознавания по сравнению с другими методами, что доказывает его практическую применимость в реальных задачах при условии большого объема поступающих данных для обучения. Кроме того, время, затраченное на обучение алгоритма, оказалось ниже, чем для методов из программного пакета MOA.



## Заключение

В данной работе рассмотрена задача обучения распознаванию образов в нестационарной генеральной совокупности. Описание генеральной совокупности построено на модели логистической регрессии. Свойство нестационарности понимается как разделяющая гиперплоскость, параметры которой изменяются во времени. В представленной постановке задачи обучения эти параметры описываются как марковские случайные процессы. Для оценивания параметров применяется байесовский подход к классификации. Нахождение их оптимальных значений в каждый момент времени осуществляется на основании процедуры динамического программирования.

Построенный алгоритм обучения распознаванию образов в нестационарной генеральной совокупности обладает линейной вычислительной сложностью относительно длины обучающей совокупности. Исследование метода на искусственных данных подтвердило его приспособляемость к происходящим в генеральной совокупности изменениям. Сравнение с методами для задач со смещением концепта из состава программного пакета Massive Online Analysis (MOA) на данных, описывающих электронные письма, показало приемлемую вычислительную эффективность предложенного метода.

## Список использованных источников

1. Айзерман М.А., Браверман Э.М., Розоноэр Л.И. Метод потенциальных функций в теории обучения машин. М.: Наука, 1970, 384 с.
2. Журавлев Ю.И., Никифоров В.В. Алгоритмы распознавания, основанные на вычислении оценок. Кибернетика, 1971, № 3.
3. Вапник В.Н. Восстановление зависимостей по эмпирическим данным. М.: Наука, 1979, 447 с.
4. Vapnik V. Statistical Learning Theory. John-Wiley&Sons, Inc. 1998.
5. Wiener N. Extrapolation, Interpolation, and Smoothing of Stationary Random Time Series with Engineering Applications. Technology Press of MIT, John Wiley&Sons, 1949, 163 p.
6. Шавловский М.Б., Красоткина О.В, Моттль В.В. Задача обучения распознаванию образов в нестационарной генеральной совокупности. Математические методы распознавания образов-13, 2007,- С. 226-230
7. Де Гроот М. Оптимальные статистические решения. М., Мир, 1974, 196 с.
8. Красоткина О. В., Моттль В. В., Турков П. А., Байесовский подход к задаче обучения распознаванию образов в нестационарной генеральной совокупности// Интеллектуализация обработки информации: 8-я международная конференция. Республика Кипр, г.Пафос, 17-24 октября 2010 г.: Сборник докладов. – М.: МАКС Пресс, 2010. – 556с. с.379-382.
9. Шавловский М.Б. Задача обучения распознаванию образов в нестационарной генеральной совокупности. Выпускная квалификационная работа магистра.- МФТИ. 2009
- 10 J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in Advances in Artificial Intelligence (Lecture Notes in Computer Science), vol. 3171. New York: Springer-Verlag, 2004, pp. 286–295.
- 11 P. Vorburger and A. Bernstein, "Entropy-based concept shift detection," in Proc. 6th Int. Conf. Data Min., 2006, pp. 1113–1118.
- 12 Widmer, G. and Kubat, M.; Learning in the presence of concept drift and hidden contexts. Machine Learning 23 (1996) 69-101.
- 13 Albert Bifet and Ricard Gavaldà. Learning from time-changing data with adaptive windowing. In SIAM International Conference on Data Mining, 2007.
- 14 Salganicoff M., Tolerating concept and sampling shift in lazy learning using prediction error context switching, AI Review, Special Issue on Lazy Learning, 11 (1-5), 1997, 133-155.
15. Maloof, M. A. and Michalski, R. S.; Incremental learning with partial instance memory. Artificial Intelligence 154 (2004) 95-126.
16. Black, M. and Hickey, R.; Learning classification rules for telecom customer call data under concept drift. Soft Computing - A Fusion of Foundations, Methodologies and Applications 8 (2003) 102-108.
17. Street, W.N., Kim, Y.: A streaming ensemble algorithm (SEA) for large-scale classification. In: KDD, pp. 377–382. ACM Press (2001)

- 18 Wang, H., Fan, W., Yu, P.S., Han, J.: Mining concept-drifting data streams using ensemble classifiers. In: Getoor, L., Senator, T.E., Domingos, P., Faloutsos, C. (eds.) KDD, pp. 226–235. ACM Press (2003)
19. Albert Bifet, Geoff Holmes, Bernhard Pfahringer, Richard Kirkby, Ricard Gavaldà. New ensemble methods for evolving data streams. In 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009
20. Albert Bifet, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer Accurate Ensembles for Data Streams: Combining Restricted Hoeffding Trees using Stacking. In Journal of Machine Learning Research — Proceedings Track 13, 225-240 (2010)
- 21 . Татарчук А.И., Сулимова В.В., Моттль В.В., Уиндридж Д. Метод релевантных потенциальных функций для селективного комбинирования разнородной информации при обучении распознаванию образов на основе байесовского подхода. // Всероссийская конференция ММРО-14.М.: МАКС Пресс, 2009.С. 188–191.
- 22 .  
Турков П.А. Разработка алгоритма обучения распознаванию образов в нестационарной генеральной совокупности данных. Выпускная квалификационная работа бакалавра.- ТулГУ. 2010.
- 23 . Красоткина О.В. Алгоритмы оценивания моделей нестационарных сигналов при наличии ограничений. Диссертация на соискание ученой степени к.ф.-м.н., М.: Тульский Государственный Университет - 2003.
24. С. М. Bishop Pattern Recognition and Machine Learning (Information Science and Statistics) // Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2006.
25. Беллман Р., Калаба Р., Динамическое программирование и современная теория управления. – М.: Наука, 1969, - 118.
- 26 . <http://archive.ics.uci.edu/ml/datasets/Spambase>, UCI Repository, Spambase Data Set
27. A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, MOA: Massive Online Analysis <http://sourceforge.net/projects/moa-datastream/>. Journal of Machine Learning Research (JMLR), 2010.
28. Manuel Baena-Garcia, Jose del Campo-Avila, Raul Fidalgo, Albert Bifet, Ricard Gavaldà, and Rafael Morales-Bueno. Early drift detection method. In Fourth International Workshop on Knowledge Discovery from Data Streams, 2006.