Автоматическое определение количества компонент в EM-алгоритме восстановления смеси нормальных распределений*

©2009г. Д. П. Ветров[1], Д. А. Кропотов[2], А. А. Осокин[1]
[1] 119992, Москва, Воробьевы горы, 1, МГУ ф-т ВМиК
[2] 119333, Москва, ул. Вавилова, д. 40, ВЦ РАН
vetrovd@yandex.ru, dmitry.kropotov@gmail.com, osokin.anton@gmail.com

Поступила в редакцию 23.07.2009 г.

Классический EM-алгоритм восстановления смеси нормальных распределений не позволяет определять количество компонент смеси. В работе предлагается алгоритм автоматического определения числа компонент ARD EM, основанный на методе релевантных векторов. Идея алгоритма состоит в использовании на начальном этапе заведомо избыточного количества компонент смеси с дальнейшим определением релевантных компонент с помощью максимизации обоснованности. Эксперименты на модельных задачах показывают, что количество найденных кластеров либо совпадает с истинным, либо немного превосходит его. Кроме того, кластеризация с помощью ARD EM оказывается ближе к истинной, чем у аналогов, основанных на скользящем контроле и принципе минимальной длины описания. Библ. 14. Табл. 4.

Ключевые слова: распознавание образов, восстановление плотностей, кластерный анализ, определение числа кластеров, EM-алгоритм, байесовское обучение, автоматическое определение релевантности

1. Введение

В данной работе рассматривается EM-алгоритм разделения смесей нормальных распределений (см. [1]) как инструмент для решения задачи кластеризации. Классический EM-алгоритм решает данную задачу при фиксированном числе кластеров. Число кластеров — структурный параметр, настройка которого является сложной задачей. Целью данного исследования является разработка метода автоматического определения числа кластеров.

Одним из возможных решений данной проблемы является применение к выборке EM-алгоритма для различного числа компонент смеси с последующим выбором лучшего решения по некоторому критерию качества. Недостатками данного подхода являются его высокая вычислительная сложность и трудности, возникающие при выборе критерия качества.

*Работа выполнена при финансовой поддержке РФФИ (коды проектов 08-01-00405, 08-01-90427, 08-01-90016, 09-01-12060, 07-01-00211) и программы ОМН РАН №02

В данной работе для определения числа компонент смеси используются методы т.н. байесовского обучения (см. [2]), которые в настоящее время широко применяются для решения задач выбора модели. В частности, авторами предложено применить прием, аналогичный использованному в методе релевантных векторов (RVM) (см. [3]). Его идея заключается в использовании индивидуальных коэффициентов регуляризации для каждого фактора, необходимость учета которого в модели не очевидна. В методе релевантных векторов в качестве таковых факторов выступают обобщенные признаки (базисные функции). В данной работе к факторам предложено отнести кластеры (компоненты смеси). Настройка индивидуальных коэффициентов регуляризации осуществляется путем максимизации т.н. обоснованности (правдоподобия модели, см. [4]). На основе этой процедуры разработана модификация EM-алгоритма, позволяющая, в отличие от аналогов, за один проход определять не только параметры компонент смеси, но и их количество. В работе проведено сравнение предложенного подхода с более простым вариантом определения числа кластеров путем последовательного запуска EM-алгоритма с возрастающим количеством компонент и последующим подсчетом обоснованности. Среди других альтернатив рассмотрен способ оценки числа кластеров с помощью минимизации длины описания (см. [5]), а также с помощью скользящего контроля. Кроме того, проведено сравнение с «идеальным» методом, который использует информацию об истинном числе компонент смеси.

Дальнейшее содержание работы состоит из трех разделов. В разд. 2 приводится описание классического EM-алгоритма как исходного объекта для дальнейших исследований. В разд. 3 описывается теоретическое построение предлагаемого алгоритма ARD EM. Также в нем приведена краткая характеристика существующих аналогов. Разд. 4 посвящен результатам экспериментов, проведенных для оценки качества работы полученного алгоритма.

2. Классический EM-алгоритм

2.1. Разделение смеси распределений

Определение 1. Смесью распределений для действительной многомерной случайной величины $\boldsymbol{x} \in \mathbb{R}^d$ будем называть распределение с плотностью

$$p(\boldsymbol{x}) = \sum_{j=1}^{K} w_j p_j(\boldsymbol{x}), \sum_{j=1}^{K} w_j = 1, w_j \geqslant 0, j = 1, \ldots, K. \tag{2.1}$$

Здесь $K \in \mathbb{N}$ — количество компонент в смеси, $w_j \in \mathbb{R}$ — веса компонент, а $p_j(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$ — плотности распределения компонент смеси.

Пусть дана выборка $X = \{\boldsymbol{x}_n\}_{n=1}^N$, $\boldsymbol{x}_n \in \mathbb{R}^d$, N — число объектов в выборке. Пусть далее функции распределения компонент смеси заданы параметрически $p_j(\boldsymbol{x}) = p(\boldsymbol{x}|\boldsymbol{\theta}_j)$, $j = 1, \ldots, K$. Тогда задача восстановления смеси распределений заключается в определении по выборке X параметров компонент $\{\boldsymbol{\theta}_j\}_{j=1}^K$ и весов $\{w_j\}_{j=1}^K$. В дальнейшем будем обозначать совокупность всех параметров смеси через $\Xi = \{\Theta, \boldsymbol{w}\} = \{\boldsymbol{\theta}_j, w_j\}_{j=1}^K$.

Для решения задачи восстановления смеси воспользуемся методом максимального правдоподобия:

$$\Xi_{ML} = \arg\max_{\Xi} p(X|\Xi) = \arg\max_{\Xi} \prod_{n=1}^{N} p(\boldsymbol{x}_n|\Xi) \tag{2.2}$$

Переходя к логарифму правдоподобия получаем следующую задачу условной оптимизации:

$$L(X,\Xi) = \log p(X|\Xi) = \sum_{n=1}^{N} \log \sum_{j=1}^{K} w_j p(\boldsymbol{x}_n|\theta_j) \to \max_{\Xi}, \tag{2.3}$$

$$\sum_{j=1}^{K} w_j = 1, w_j \geqslant 0, j = 1 \dots K. \tag{2.4}$$

Функционал (2.3) имеет вид «логарифм суммы» и сложен для прямой оптимизации. Наиболее распространенным путем решения этой задачи является введение скрытых переменных с дальнейшим применением EM-алгоритма максимизации неполного правдоподобия (т.е. правдоподобия при наличии скрытых переменных). Известно (см. [6]), что EM-алгоритм для задачи разделения смесей распределения имеет ряд преимуществ перед другими методами оптимизации, такими как проекция градиента и Ньютоновские алгоритмы. Поэтому в дальнейшем в работе везде для решения задачи разделения смеси будет использоваться EM-алгоритм.

Рассмотрим вероятностную модель, эквивалентную (2.1). Для этого представим процесс генерации объекта из смеси распределений в два этапа: сначала с вероятностями пропорциональными весам $\boldsymbol{w}$ выбирается одна компонента смеси, а затем из этой компоненты генерируется объект $\boldsymbol{x}$. Формально для каждого объекта $\boldsymbol{x}$ вводится случайная переменная $\boldsymbol{t}$, определяющая соответствующий $\boldsymbol{x}$ номер компоненты смеси:

$$\boldsymbol{t} = (t_1, \dots, t_K),\ t_j = \begin{cases} 1, & \text{если объект } \boldsymbol{x} \text{ взят из } j\text{-ой компоненты смеси} \\ 0, & \text{иначе} \end{cases},\ \sum_{j=1}^{K} t_j = 1.$$

При этом

$$p(\boldsymbol{t}) = \prod_{j=1}^{K} w_j^{t_j}, \tag{2.5}$$

$$p(\boldsymbol{x}|\boldsymbol{t}) = \prod_{j=1}^{K} [p_j(\boldsymbol{x})]^{t_j}. \tag{2.6}$$

Легко показать, что маргинальное распределение $p(\boldsymbol{x})$ в вероятностной модели (2.5)-(2.6) совпадает с распределением (2.1). В этом смысле модели (2.5)-(2.6) и (2.1) эквивалентны.

2.2. Идея EM-алгоритма

EM-алгоритм позволяет максимизировать правдоподобие в вероятностных моделях со скрытыми переменными (см. [1]). Пусть имеется некоторая вероятностная модель, в которой часть переменных X известна, часть переменных T не наблюдается, а также имеется набор параметров Ξ. Задача состоит в отыскании параметров Ξ путем максимизации правдоподобия:

$$p(X|\Xi) = \int p(X,T|\Xi)dT = \int p(X|T,\Xi)p(T|\Xi)dT \to \max_{\Xi}$$

EM-алгоритм представляет собой итерационную схему, состоящую из двух шагов. В начале выбирается некоторое значение параметров Ξ_{old}. Далее на первом шаге (E-шаг, ожидание)

вычисляется апостериорное распределение на скрытые компоненты T при фиксированном значении параметров Ξ_{old}:

$$p(T|X,\Xi_{old}) = \frac{p(X,T|\Xi_{old})}{\int p(X,T|\Xi_{old})dT}. \tag{2.7}$$

Затем на втором шаге (M-шаг, максимизация) логарифм полного правдоподобия усредняется по полученному апостериорному распределению и максимизируется для поиска новых значений параметров Ξ_{new}:

$$\Xi_{new} = \arg\max_{\Xi} \mathbb{E}_{T|X,\Xi_{old}} \log p(X,T|\Xi) \tag{2.8}$$

Шаги E и M повторяются в цикле до сходимости. Можно показать (см. [1]), что на каждой итерации значение логарифма правдоподобия не уменьшается. Таким образом, EM-алгоритм позволяет находить локальный максимум правдоподобия.

2.3. Применение EM-алгоритма для разделения смеси распределений

Рассмотрим применение EM-алгоритма для вероятностной модели (2.5)-(2.6), в которой в качестве скрытых переменных выступают номера компонент смеси $T = \{\boldsymbol{t}_n\}_{n=1}^N$, соответствующие объектам выборки $X = \{\boldsymbol{x}_n\}_{n=1}^N$. E-шаг (2.7) производится следующим образом:

$$\gamma_{nj} = \mathbb{E}_{t_{nj}|\boldsymbol{x}_n,\Xi} t_{nj} = p(t_{nj}|\boldsymbol{x}_n,\Xi) = \frac{w_j p(\boldsymbol{x}_n|\boldsymbol{\theta}_j)}{\sum_{k=1}^K w_k p(\boldsymbol{x}_n|\boldsymbol{\theta}_k)} \tag{2.9}$$

Заметим, что для дальнейшего вычисления математического ожидания логарифма полного правдоподобия на M-шаге в данном случае достаточно знать лишь апостериорное распределение на отдельные компоненты t_{nj}.

Задача оптимизации на M-шаге (2.8) записывается следующим образом:

$$\sum_{n=1}^N \sum_{j=1}^K \gamma_{nj} \left(\log w_j + \log p(\boldsymbol{x}_n|\boldsymbol{\theta}_j)\right) \to \max_{\Xi}$$

2.4. Случай смеси нормальных распределений

В дальнейшем в качестве основного объекта исследований будет рассматриваться смесь нормальных распределений, т.е. в качестве компонент смеси выбираются следующие:

$$p(\boldsymbol{x}|\boldsymbol{\theta}_j) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j,\Sigma_j) = \frac{1}{(2\pi)^{\frac{d}{2}}} \frac{1}{\sqrt{|\Sigma_j|}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_j)\right), \quad \boldsymbol{\theta}_j \equiv (\boldsymbol{\mu}_j,\Sigma_j).$$

Здесь $\boldsymbol{\mu}_j \in \mathbb{R}^d$ — вектор математических ожиданий для j-ой компоненты, а $\Sigma_j \in \mathbb{R}^{d\times d}$ — произвольная симметричная неотрицательно определенная матрица ковариации для j-ой компоненты смеси.

В случае нормальных распределений M-шаг EM-алгоритма выполняется аналитически:

$$w_j = \frac{\sum_{n=1}^N \gamma_{nj}}{N}, \tag{2.10}$$

$$\boldsymbol{\mu}_j = \frac{1}{N w_j} \sum_{n=1}^N \gamma_{nj} \boldsymbol{x}_n, \tag{2.11}$$

$$\Sigma_j = \frac{1}{N w_j} \sum_{n=1}^N \gamma_{nj} (\boldsymbol{x}_n - \boldsymbol{\mu}_j)(\boldsymbol{x}_n - \boldsymbol{\mu}_j)^T. \tag{2.12}$$

3. EM-алгоритм с автоматическим определением числа компонент

Применение классического EM-алгоритма для восстановления смеси нормальных распределений требует задания количества компонент K. В том случае, если K не известно, то возникает задача автоматического выбора количества компонент смеси по данным. Эта задача не может быть решена простым включением K в набор параметров Ξ с дальнейшим поиском параметров по максимуму правдоподобия. Действительно, чем больше значение K, тем, вообще говоря, больше значение правдоподобия, т.к. более гибкая модель может лучше объяснить имеющиеся данные. В частности, при $K = N$ мы приходим к вырожденному решению, когда значение правдоподобия равно бесконечности, а компоненты смеси имеют центры в объектах выборки с нулевыми матрицами ковариации. Выбор числа кластеров является частным случаем проблемы автоматического выбора модели в задачах машинного обучения, заключающейся в наличии ряда параметров (обычно называемых структурными параметрами или параметрами модели), которые не могут быть автоматически определены в рамках классических алгоритмов обучения.

3.1. Обзор методов определения параметров модели

Скользящий контроль

В настоящее время скользящий контроль является наиболее надежным средством настройки структурных параметров в моделях машинного обучения. Процедура скользящего контроля (кросс-валидации) заключается в последовательном исключении части объектов из обучающей выборки, обучении на оставшихся объектах и вычислении качества полученной модели для исключенных объектов. Таким образом эмулируется наличие тестовой выборки, которая не участвует в обучении. Структурные параметры настраиваются путем максимизации критерия качества модели на скользящем контроле. В данном случае в качестве критерия соответствия модели данным выбирается значение логарифма правдоподобия (2.3). Итоговое значение критерия определяется как сумма логарифмов правдоподобия для каждой рассматриваемой части выборки.

Принцип минимальной длины описания (MDL)

Одним из общих принципов выбора модели для некоторого набора данных является принцип минимальной длины описания (см. [7]), который формулируется следующим образом: следует выбрать ту модель, которая позволяет описать данные и порождающую их модель наиболее коротко. Правила построения описания (и оценка его длины) будут являться формализацией понятия «адекватности описания». При использовании принципа минимальной длины описания предполагается, что чем сложнее решающее правило (т.е. чем длиннее его описание), тем хуже его обобщающая способность. Современные исследования (в частности, boosting, см. [8]) показывают, что это далеко не всегда так.

В работе [5] получен критерий, основанный на MDL, который позволяет определить количество компонент в смеси распределений:

$$\mathrm{MDL} = -\sum_{j=1}^{K} N_j \log\left(\frac{N_j^2}{\det \Sigma_j}\right) + K(d^2 + 3d + 2)\log(N)/2. \tag{3.1}$$

Здесь N_j — количество объектов, отнесенных к j-й компоненте смеси. В разд. 3 проведено сравнение алгоритма ARD EM с алгоритмом, основанным на данном критерии.

Информационный критерий Акаике

В начале 70-х гг. Акаике получил несмещенную оценку правдоподобия тестовой выборки, выраженную через правдоподобие обучающей выборки при использовании оценки максимального правдоподобия, полученной по обучающей выборке (см. [9]). Максимизация оценки правдоподобия тестовой выборки эквивалентна минимизации информационного критерия Акаике

$$\mathrm{AIC} = 2M - 2\log p(X|\Xi);$$

где M — число настраиваемых параметров.

Критерий является (весьма грубым) приближением более сложного выражения, часто не поддающегося аналитическому вычислению. Значение критерия может расцениваться лишь как приблизительная характеристика обобщающей способности полученного решающего правила. Более того, критерий разумно использовать, когда все M настраиваемых параметров оказывают примерно одинаковое влияние на вид решающего правила, например, входят в него линейно. В поставленной задаче разделения гауссовской смеси параметры по-разному входят в решающее правило, поэтому применение критерия Акаике некорректно — в задаче кластеризации его значение не будет связано с правдоподобием тестовой выборки.

3.2. Идея ARD EM

В данной работе для автоматического выбора числа компонент предложено использовать метод байесовского обучения. Известно, что во многих случаях он позволяет найти априорное распределение на оцениваемые параметры, наиболее адекватное наблюдаемым данным (см. [4]). В частности, использование т.н. априорного распределения с автоматическим определением релевантности (ARD prior) позволяет эффективно отсекать избыточные параметры, предотвращая переобучение модели (см. [3]). Воспользуемся аналогичным приемом. Установим начальное число компонент смеси $K = \sqrt{N}$. Будем считать, что это явно завышенное число кластеров. Введем априорное распределение на веса смеси $\boldsymbol{w}$

$$p(w_i|\alpha_i) = \sqrt{\frac{\alpha_i}{2\pi}} \exp\left(-\frac{1}{2} w_i^2 \alpha_i\right) \tag{3.2}$$

Такое априорное распределение соответствует независимой регуляризации каждого веса w_i со своим параметром регуляризации $\alpha_i \geqslant 0$.

Обозначим $\Xi_{MP} = (\Theta_{MP}, \boldsymbol{w}_{MP}) = \arg\max_{(\Theta,\boldsymbol{w})} p(X|\Theta,\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})$. Модификация EM-алгоритма, необходимая для оптимизации такого функционала, будет рассмотрена в разделе *3.4*. Для подбора параметров модели α_i воспользуемся идеей максимизации т.н. обоснованности (см. [4]):

$$p(X|\Theta_{MP}(\boldsymbol{\alpha}),\boldsymbol{\alpha}) = \int p(X|\Theta_{MP}(\boldsymbol{\alpha}),\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w} \to \max_{\boldsymbol{\alpha}}.$$

Мы предполагаем, что в процессе такой оптимизации для выбранных функций правдоподобия и априорного распределения значительная часть элементов $\alpha_i \to +\infty$, а соответствующие им веса w_i будут стремиться к нулю. Компоненты, определяемые этими параметрами, перестанут влиять на правдоподобие, и их можно будет удалить из модели, сократив число кластеров.

3.3. Вычисление обоснованности

Для оценки параметров априорного распределения $\boldsymbol{\alpha}$ будем оптимизировать величину:

$$p(X|\Theta_{MP}(\boldsymbol{\alpha}),\boldsymbol{\alpha}) = \int_{\mathcal{W}} p(X|\Theta_{MP}(\boldsymbol{\alpha}),\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w} \to \max_{\boldsymbol{\alpha}}$$

Заметим, что интеграл берется не по всему пространству $\boldsymbol{w} \in \mathbb{R}^K$, а по симплексу

$$\mathcal{W} = \left\{ \boldsymbol{w} : \sum_{j=1}^{K} w_j = 1, w_j \geqslant 0 \right\}.$$

Подынтегральное выражение представляет из себя произведение взвешенной суммы гауссиан и нормального распределения. Этот интеграл может быть вычислен аналитически, но при этом количество слагаемых растет экспоненциально и при большом количестве компонент не поддается обработке. Поэтому воспользуемся приближением Лапласа (см. [2]) — приблизим подынтегральную функцию гауссианой, интеграл от которой легко берется. Учитывая легкость хвостов гауссианы (равно как и исходной функции), заменим интеграл по симплексу интегралом по линейному многообразию

$$\mathcal{M} = \left\{ \sum_{j=1}^{K} w_j = 1 \right\}.$$

Тогда оценка логарифма обоснованности с помощью приближения Лапласа примет вид[1]

$$\log p(X|\Theta_{MP}(\boldsymbol{\alpha}),\boldsymbol{\alpha}) = \log p(X|\Theta_{MP}(\boldsymbol{\alpha}),\boldsymbol{w}_{MP})p(\boldsymbol{w}_{MP}|\boldsymbol{\alpha}) - \frac{1}{2}\log\det H_{pr} + \text{Const}, \tag{3.3}$$

где H_{pr} — проекция гессиана $H = \nabla_{\boldsymbol{w}}\nabla_{\boldsymbol{w}} \log p(X|\Theta_{MP},\boldsymbol{w}_{MP})p(\boldsymbol{w}_{MP}|\boldsymbol{\alpha})$, вычисленного в точке $\boldsymbol{w}_{MP}$, на многообразии $\mathcal{M}$, подсчитанная в произвольном ортонормированном базисе подпространства, порождающего $\mathcal{M}$. Заметим, что значение $\log\det H_{pr}$ не зависит от конкретного выбора ортонормированного базиса в этом подпространстве.

Теорема 1. *Необходимым условием максимума выражения обоснованности* (3.3) *является выполнение следующей системы равенств*

$$\frac{1}{2\alpha_k} - \frac{1}{2}w_{MP,k}^2 - \frac{1}{2}\frac{\partial \log\det H_{pr}}{\partial \alpha_k} = 0, k = 1\ldots K. \tag{3.4}$$

Доказательство. Перепишем выражение (3.3):

$$\log p(X|\Theta_{MP},\boldsymbol{\alpha}) = \log p(X|\Theta_{MP},\boldsymbol{w}_{MP}) + \log p(\boldsymbol{w}_{MP}|\boldsymbol{\alpha}) - \frac{1}{2}\log\det H_{pr} =$$
$$= \log p(X|\Theta_{MP},\boldsymbol{w}_{MP}) + k\log\sqrt{\frac{2}{\pi}} + \frac{1}{2}\sum_{j=1}^{k}\log\alpha_i - \frac{1}{2}\sum_{j=1}^{k} w_{MP,j}^2\alpha_j - \frac{1}{2}\log\det H_{pr}.$$

Вычислим производную логарифма обоснованности по $\boldsymbol{\alpha}$:

$$\frac{\partial(\log p(X|\Theta_{MP},\boldsymbol{\alpha}))}{\partial\alpha_k} = \frac{1}{2\alpha_k} - \frac{1}{2}w_{MP,k}^2 - \frac{1}{2}\frac{\partial\log\det H_{pr}}{\partial\alpha_k}, \quad k = 1\ldots K.$$

Необходимым условием максимума выражения (3.3) является равенство нулю его производной по $\boldsymbol{\alpha}$. Теорема доказана. □

[1]Мы воспользовались фактом, что $\boldsymbol{w}_{MP} \in \mathcal{M}$ (см. раздел *3.4.*) и формулой $\int f(\boldsymbol{x})d\boldsymbol{x} = \frac{\left(\sqrt{2\pi}\right)^m f(\boldsymbol{\mu})}{\sqrt{\det H}}$ при $f(\boldsymbol{x}) = \exp(-0.5(\boldsymbol{x}-\boldsymbol{\mu})^T H(\boldsymbol{x}-\boldsymbol{\mu}))$, $\boldsymbol{x} \in \mathbb{R}^m$.

Для того чтобы воспользоваться результатом теоремы 1, необходимо уметь вычислять значение производной $\frac{\partial}{\partial \alpha_k} \log\det H_{pr}$. Рассмотрим произвольный базис в подпространстве $\mathcal{L} = \left\{\boldsymbol{w}\middle|\sum_{k=1}^K w_k = 0\right\}$, порождающем многообразие $\mathcal{M}$. Обозначим $S \in \mathbb{R}^{K\times(K-1)}$ матрицу, в столбцах которой записаны координаты этого базиса. Пусть матрица $T \in \mathbb{R}^{(K-1)\times(K-1)}$ задает переход из этого базиса к некоторому ортонормированному базису $\mathcal{L}$. Тогда справедливы равенства

$$\log\det H_{pr} = \log\det(T^T S^T H S T) = 2\log\det T + \log\det(S^T H S).$$

Отсюда следует, что производная логарифма определителя H_{pr} по $\boldsymbol{\alpha}$ не зависит от выбора матриц T и S. В то же время, варьируя матрицу S возможно получать различные итерационные формулы для решения системы уравнений (3.4).

Введем обозначение $G \in \mathbb{R}^{N\times K}, G_{nj} = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_j, \Sigma_j)$.

Лемма 1. *Справедлива формула*

$$H = (G^T \Phi G + A), \tag{3.5}$$

где $A = \operatorname{diag}(\boldsymbol{\alpha})$, $\Phi = \operatorname{diag}\left(\frac{1}{\left(\sum_{j=1}^k w_j G_{1j}\right)^2}, \ldots, \frac{1}{\left(\sum_{j=1}^k w_j G_{Nj}\right)^2}\right) \in \mathbb{R}^{N\times N}$.

Доказательство. Вычислим гессиан логарифма функции правдоподобия выборки в точке максимума.

$$\log p(X|\Theta, \boldsymbol{w}) = \sum_{n=1}^N \log\left(\sum_{j=1}^K w_j G_{nj}\right),$$

$$\frac{\partial}{\partial w_s}\log p(X|\Theta, \boldsymbol{w}) = \sum_{n=1}^N \frac{G_{ns}}{\sum_{j=1}^K w_j G_{nj}}, \text{где } s = 1, \ldots, K,$$

$$\frac{\partial^2}{\partial w_s \partial w_q}\log p(X|\Theta, \boldsymbol{w}) = -\sum_{n=1}^N \frac{G_{ns}G_{nq}}{\left(\sum_{j=1}^K w_j G_{nj}\right)^2}, \text{где } s = 1, \ldots, K,\ q = 1, \ldots, K.$$

Таким образом,

$$\nabla\nabla \log p(X|\Theta, \boldsymbol{w}_{MP}) = -G^T \Phi G, \text{где } \Phi = \operatorname{diag}\left(\frac{1}{\left(\sum_{j=1}^K w_j G_{nj}\right)^2}\right). \tag{3.6}$$

Вычислим гессиан логарифма плотности априорного распределения в точке $\boldsymbol{w}_{MP}$:

$$\nabla\nabla \log p(\boldsymbol{w}_{MP}|\boldsymbol{\alpha}) = \nabla\nabla\left(-\frac{1}{2}\sum_{j=1}^K w_j^2 \alpha_j\right) = \operatorname{diag}(-\boldsymbol{\alpha}) = -A. \tag{3.7}$$

Объединяя (3.6) и (3.7), окончательно получаем (3.5). □

Рассмотрим матрицу S, равную

$$S = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \\ -1 & -1 & \cdots & -1 \end{pmatrix} \in \mathbb{R}^{K\times(K-1)}.$$

Как было отмечено выше, с учетом леммы 1 имеем

$$\frac{\partial}{\partial\alpha_j}\log\det H_{pr} = \frac{\partial}{\partial\alpha_j}\log\det S^THS = \frac{\partial}{\partial\alpha_j}\log\det\left(S^TG^T\Phi GS + S^TAS\right).$$

Применим формулу $\frac{\partial}{\partial \boldsymbol{x}}\log\det M = \operatorname{tr}\left(M^{-1}\frac{\partial M}{\partial \boldsymbol{x}}\right)$ к матрице S^THS

$$\frac{\partial}{\partial\alpha_j}\log\det S^THS = \operatorname{tr}\left((S^THS)^{-1}\frac{\partial S^THS}{\partial\alpha_j}\right) = \operatorname{tr}\left((S^THS)^{-1}\frac{\partial S^TAS}{\partial\alpha_j}\right)$$

$$\text{Если } j = 1,\ldots,K-1, \text{ то } \frac{\partial S^TAS}{\partial\alpha_j} = \begin{array}{c} \\ j \end{array}\begin{pmatrix} & & \overset{j}{} & & \\ 0 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \cdots & 0 \end{pmatrix}$$

$$\text{Если } j = K, \text{ то } \frac{\partial S^TAS}{\partial\alpha_j} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}$$

Окончательно получаем

$$\frac{\partial}{\partial\alpha_j}\log\det H_{pr} = \begin{cases} j = 1,\ldots,K-1, & ((S^THS)^{-1})_{jj} \\ j = K, & \sum_{l,m=1}^{K}((S^THS)^{-1})_{lm} \end{cases}$$

С учетом вида выражения для $\frac{\partial}{\partial\alpha_j}\log\det H_{pr}$ и системы (3.4), получим итеративные формулы для пересчета $\boldsymbol{\alpha}$

$$\alpha_j^{new} = \frac{1-\alpha_j^{old}((S^THS)^{-1})_{jj}}{w_{MP,j}^2}, \quad j = 1,\ldots,K-1 \tag{3.8}$$

$$\alpha_K^{new} = \frac{1-\alpha_K^{old}\sum_{l,m=1}^{K}((S^THS)^{-1})_{lm}}{w_{MP,K}^2}\,. \tag{3.9}$$

3.4. EM-алгоритм максимизации регуляризованного правдоподобия

В процессе оценки и оптимизации обоснованности требуется вычисление точки максимума регуляризованного правдоподобия Ξ_{MP}:

$$\Xi_{MP} = (\Theta_{MP}, \boldsymbol{w}_{MP}) = \arg\max_{(\Theta,\boldsymbol{w})}[\log p(X|\Theta,\boldsymbol{w}) + \log p(\boldsymbol{w}|\boldsymbol{\alpha})]$$

Рассмотрим вопрос модификации стандартного EM-алгоритма максимизации правдоподобия для смеси нормальных распределений с E-шагом (2.9) и M-шагом (2.10)-(2.12) на случай максимизации регуляризованного правдоподобия с априорным распределением (3.2).

С точки зрения общего вида EM-алгоритма E-шаг (2.7) не претерпевает изменений, а M-шаг (2.8) видоизменяется следующим образом:

$$\Xi_{new} = \arg\max_{\Xi}\left[\mathbb{E}_{T|X,\Xi_{old}}\log p(X,T|\Xi) + \log p(\Xi)\right] \tag{3.10}$$

Здесь $p(\Xi)$ — априорное распределение на набор оцениваемых параметров смеси Ξ. Предполагается, что в итерационном процессе ЕМ-алгоритма распределение $p(\Xi)$ не меняется.

Применительно к задаче разделения смеси с регуляризацией на веса $\boldsymbol{w}$, М-шаг принимает вид:

$$L(X,\Xi) = \sum_{n=1}^{N} \log \sum_{j=1}^{K} w_j p(\boldsymbol{x}_n|\theta_j) - \frac{1}{2}\sum_{j=1}^{K} w_j^2 \alpha_j \to \max_{\Xi}, \tag{3.11}$$

$$\sum_{j=1}^{K} w_j = 1,\ w_j \geqslant 0,\ j = 1 \ldots K. \tag{3.12}$$

Теорема 2. *Необходимым условием максимума в задаче* (3.11)-(3.12) *является выполнение следующей системы уравнений*

$$\sum_{n=1}^{N} \frac{w_j p(\boldsymbol{x}_n|\theta_j)}{\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n|\theta_k)} - w_j^2\alpha_j + w_j\left(\sum_{k=1}^{K} w_k^2\alpha_k - m\right) = 0, \qquad j = 1,\ldots,K. \tag{3.13}$$

Доказательство. Заметим сразу, что априорное распределение зависит только от вектора весов $\boldsymbol{w}$. Следовательно, решение данной задачи для параметров компонент смеси $\boldsymbol{\theta}_j$ находится по формулам из классического ЕМ-алгоритма (2.11) и (2.12).

Для поиска решения по $\boldsymbol{w}$ воспользуемся методом множителей Лагранжа:

$$\Lambda(X,\Xi) = L(X,\Xi) + \lambda\left(\sum_{k=1}^{K} w_k - 1\right).$$

Приравнивая к нулю частные производные функции Λ, получаем:

$$\frac{\partial\Lambda}{\partial w_j} = \sum_{n=1}^{N} \frac{p(\boldsymbol{x}_n|\theta_j)}{\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n|\theta_k)} - w_j\alpha_j + \lambda = 0, \tag{3.14}$$

Умножим левую и правую части равенства (3.14) на w_j:

$$w_j\frac{\partial\Lambda}{\partial w_j} = \sum_{n=1}^{N} \frac{w_j p(\boldsymbol{x}_n|\theta_j)}{\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n|\theta_k)} - w_j^2\alpha_j + \lambda w_j = 0. \tag{3.15}$$

Просуммируем уравнения (3.15) по j от 1 до K и переставим порядок суммирования в левой части:

$$\sum_{n=1}^{N}\sum_{j=1}^{K} \frac{w_j p(\boldsymbol{x}_n|\theta_j)}{\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n|\theta_k)} - \sum_{j=1}^{K} w_j^2\alpha_j = -\lambda\sum_{j=1}^{K} w_j.$$

Заметим, что $\sum\limits_{j=1}^{K} w_j = 1$. Отсюда получаем, что $\lambda = -m + \sum\limits_{s=1}^{K} w_s^2\alpha_s$. Подставляя найденное значение λ в (3.15) получаем окончательную формулу (3.13). □

Обозначив в формуле (3.13) w_j при λ за новое значение веса, а все остальные w_k за старые значения, получим итеративную формулу пересчета весов $\boldsymbol{w}$:

$$w_{j,new} = \frac{\sum_{n=1}^{N} \frac{w_{j,old} p(\boldsymbol{x}_n|\theta_j)}{\sum_{k=1}^{K} w_{k,old} p(\boldsymbol{x}_n|\theta_k)} - w_{j,old}^2\alpha_j}{m - \sum_{k=1}^{K} w_{k,old}^2\alpha_k} = \frac{\sum_{n=1}^{N}\gamma_{nj} - w_{j,old}^2\alpha_j}{m - \sum_{k=1}^{K} w_{k,old}^2\alpha_k},\ j = 1\ldots K. \tag{3.16}$$

Заметим, что из выражения (3.15) итерационные формулы пересчета $\boldsymbol{w}$ можно получать различными способами. Вариант (3.16) отличается тем, но он переходит в классические формулы пересчета (2.9) при $\alpha_j \to 0$.

3.5. Алгоритм ARD EM

Вход: Выборка $X = \{\boldsymbol{x}_n\}_{n=1}^N$, $\boldsymbol{x}_n \in \mathbb{R}^d$

Выход: Количество компонент K, веса $\boldsymbol{w}$ и параметры $\{\boldsymbol{\mu}_j\}_{j=1}^K$ и $\{\Sigma_j\}_{j=1}^K$ для компонент смеси нормальных распределений, где

$$p(\boldsymbol{x}|\boldsymbol{w}, \{\boldsymbol{\mu}_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K) = \sum_{j=1}^K w_j \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \Sigma_j)$$

1: {Инициализация}
$K := \sqrt{N}$; {Начальное количество компонент}
$(\boldsymbol{w}, \{\boldsymbol{\mu}_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K) = EM(X, K)$; {Компоненты — при помощи классического EM-алгоритма}
$\alpha_j := 1, j = 1 \dots K$; {Параметры регуляризации}
AlphaBound:= 10^3;
WeightBound:= 10^{-3};
NumberOfIterations:= 100;
2: **для** iteration= $1, \dots,$NumberOfIterations
3: $(\boldsymbol{w}, \{\boldsymbol{\mu}_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K, G) = EM_{new}(X, K, \boldsymbol{\alpha}, \boldsymbol{w}, \{\boldsymbol{\mu}_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K)$;
{EM_{new} — модификация EM-алгоритма, описанная в разделе *3.4.*, в качестве начального приближения для весов и параметров компонент смеси берутся соответственно $\boldsymbol{w}, \{\boldsymbol{\mu}_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K$; G — матрица значений плотностей компонент на каждом из объектов выборки}
4: $\Phi := \operatorname{diag} \frac{1}{\left(\sum_{j=1}^K w_j N_{ij}\right)^2}$;
$A := \operatorname{diag}(\boldsymbol{\alpha})$;
$H := G^T \Phi G + A$;
$\Sigma := \left(S^T H S\right)^{-1}$;
5: **для** $j = 1, \dots, K$
6: **если** $j < K$ **то**
7: $\alpha_j = \frac{1 - \alpha_j \Sigma_{jj}}{w_j^2}$;
8: **иначе**
9: $\alpha_K = \frac{1 - \alpha_K \sum_{l,m=1}^K \Sigma_{lm}}{w_K^2}$;
10: **если** $\alpha_j >$AlphaBound или $w_j <$WeightBound **то**
11: Удаляем j-ю компоненту смеси

3.6. Упрощенный байесовский критерий

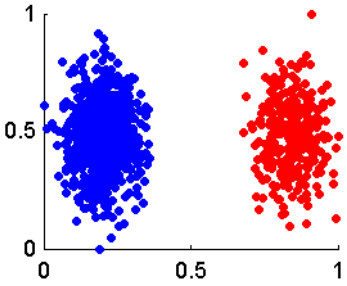
Классический EM-алгоритм разделения гауссовской смеси на k компонент можно рассматривать как частный случай изложенного в разделе *3.4.* алгоритма оптимизации регуляризованного правдоподобия при $\alpha_1 = \dots = \alpha_k = 0$, $\alpha_{k+1} = \dots = \alpha_K = +\infty$. В качестве упрощенного байесовского критерия будем считать правдоподобие модели с фиксированным числом кластеров

$$\mathrm{BIC}(k) = \int_{\boldsymbol{w} \in \mathbb{R}^k} p(X|\Theta, \boldsymbol{w}) d\boldsymbol{w} \approx \frac{p(X|\Theta_{ML}, \boldsymbol{w}_{ML})}{\sqrt{\det\left(-\nabla\nabla \log p(X|\Theta, \boldsymbol{w})|_{\Theta=\Theta_{ML}, \boldsymbol{w}=\boldsymbol{w}_{ML}}\right)}}. \tag{3.17}$$

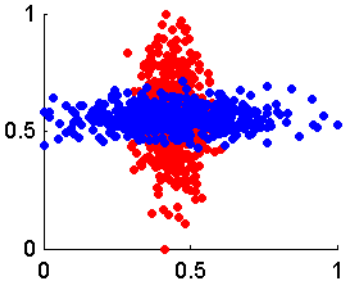
Здесь $\Theta = \{(\boldsymbol{\mu}_j, \Sigma_j)\}_{j=1}^k$, $\boldsymbol{w} \in \mathbb{R}^k$. Число кластеров определяется по максимуму упрощенного критерия

$$k_* = \arg \max_{k=1,\dots,K} \mathrm{BIC}(k).$$

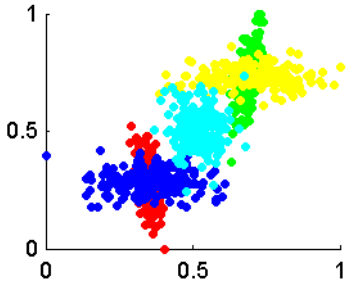
Таблица 1: Модельные задачи



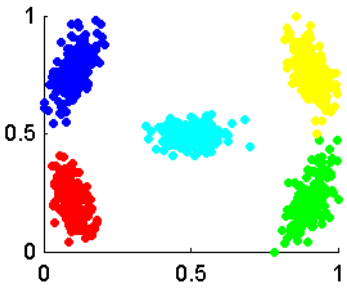
Данные №1. Два непересекающихся кластера-гауссианы.



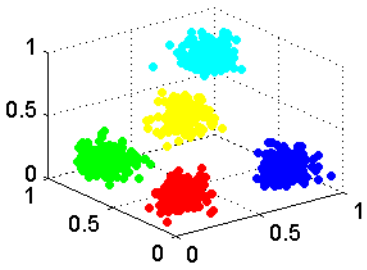
Данные №2. Два пересекающихся кластера-гауссианы.



Данные №3. Пять пересекающихся кластеров-гауссиан.



Данные №4. Пять непересекающихся кластеров-гауссиан.



Данные №5. Пять непересекающихся кластеров-гауссиан в трехмерном пространстве.

Данные №6 - хорошо разделимые кластеры-гауссианы в пятимерном пространстве.

Данные №7 - хорошо разделимые кластеры-гауссианы в десятимерном пространстве.

Данные №8 - задача Iris. 4-х мерное пространство. 150 объектов. 3 кластера.

Заметим, что в отличие от ARD EM, требующего единственного итерационного процесса для сходимости к наиболее обоснованому числу кластеров, в упрощенном байесовском критерии отсутствуют коэффициенты регуляризации и оптимальное количество кластеров находится путем последовательного запуска EM-алгоритма с увеличивающимся k. Использование упрощенного критерия и сравнение с ним алгоритма ARD EM позволит установить влияние коэффициентов регуляризации на точность кластеризации.

4. Результаты экспериментов

Для оценки качества работы рассмотренных алгоритмов были проведены эксперименты на модельных данных. Выборки, участвующие в эксперименте, указаны в таблице 1. В эксперименте принимали участие следующие алгоритмы: предлагаемый метод ARD EM, основанный на байесовском подходе, методы перебора по всем значениям k с максимизацией некоторого критерия (упрощенный байесовский критерий BIC (3.17) в BIC EM, см. раздел *3.6.*, критерий (3.1), связанный с длиной описания, в MDL EM, см. раздел *3.1.*, логарифм правдоподобия, полученный с помощью скользящего контроля, в CV EM), а также для сравнения с «идеальным» методом EM-алгоритм с истинным числом кластеров (True EM). Как было отмечено выше, EM-алгоритм позволяет находить локальный максимум правдоподобия и, в частности, результат работы метода существенно зависит от начального приближения на параметры. Чтобы снизить зависимость рассматриваемых методов от выбора начального приближения, алгоритмы запускались 10 раз из различных начальных приближений,

Таблица 2: Количество кластеров, найденных различными алгоритмами

Задача	Истинное число кластеров	ARD EM	BIC EM	MDL EM	CV EM
№1	2	2	2	9	2
№2	2	2	4	7	2
№3	5	5	6	9	8
№4	5	5	10	9	10
№5	5	5	9	8	5
№6	5	7	10	9	7
№7	5	7	10	10	8
№8	3	4	9	9	9

Таблица 3: Модифицированный показатель Ранда для различных алгоритмов

Задача	True EM	ARD EM	BIC EM	MDL EM	CV EM
№1	1	1	1	0.4436	1
№2	0.3788	0.3739	0.2799	0.1749	0.3638
№3	0.6402	0.6788	0.4208	0.4170	0.5660
№4	1	1	0.7897	0.8473	0.8970
№5	1	1	0.7967	0.8698	1
№6	1	0.9188	0.8358	0.8707	0.9087
№7	1	0.9008	0.8269	0.8269	0.8738
№8	0.9410	0.8490	0.6191	0.6191	0.6191

а в качестве ответа выбирался наилучший результат с точки зрения оптимизируемого критерия. Например, в алгоритме BIC EM выбирался тот результат оптимизации, который соответствовал наилучшему значению критерия BIC (3.17), а в CV EM на внутренних итерациях скользящего контроля EM-алгоритм запускался 10 раз из различных начальных приближений с максимизацией логарифма правдоподобия для текущей обучающей выборки. Поэтому в тех случаях, когда различные алгоритмы выбирали одно и тоже оптимальное число кластеров, их итоговые результаты кластеризации, вообще говоря, могли отличаться друг от друга, т.к. в них оптимизируются различные критерии по начальным приближениям.

В ARD EM в начале выбиралось число кластеров $\sqrt{N}$, а в алгоритмах перебора число кластеров выбиралось из множества $\{1, \ldots, \sqrt{N}\}$.

В таблице 2 приведено число кластеров, выбранное рассматриваемыми алгоритмами для каждой задачи. В таблице 3 указан модифицированный показатель Ранда (Adjusted Rand Index, см. [10]) для разбиений, сделанных алгоритмами, и истинных кластеризаций. Модифицированный показатель Ранда позволяет сравнивать близость двух разбиений и принимает значение 1 для идентичных с точностью до перестановки разбиений и значение, близкое к нулю, для независимых разбиений. Чем более похожи разбиения, тем ближе к единице значение модифицированного показателя Ранда. В таблице 4 приведено время работы различных алгоритмов.

Как видно из результатов экспериментов, кластеризация ARD EM оказывается ближе к истинной по модифицированному показателю Ранда, чем у других методов. При этом она практически не уступает по качеству EM-алгоритму с истинным числом кластеров.

Таблица 4: Время работы различных алгоритмов (в секундах)

Задача	True EM	ARD EM	BIC EM	MDL EM	CV EM
№1	0.4	59.1	1175	1159.3	9979.7
№2	2.3	51.6	994.9	1057.6	9686.5
№3	8.7	75.1	503.6	493.9	6149.1
№4	4.1	86.2	313.7	294.6	2157
№5	5.8	87.5	537.2	523.6	4159.3
№6	2.5	82.0	455.3	435.7	2527.9
№7	6.4	97.4	717.7	696.9	2983
№8	1.0	15.3	85.5	85.0	432.5

Количество кластеров, выбираемое с помощью ARD EM, также оказывается ближе к истинному по сравнению с другими методами. Также следует отметить, что время работы ARD EM существенно меньше, чем у остальных алгоритмов. Это связано с тем, что в итерационном процессе ARD EM коэффициенты регуляризации довольно быстро начинают уходить в бесконечность и, таким образом, быстро сокращается количество оптимизируемых параметров для внутреннего EM-алгоритма. В методах, связанных с перебором по числу кластеров, неизбежно возникает необходимость многократной оптимизации для больших k, что приводит к существенному росту временных затрат. Одним из возможных способов снижения временных затрат для алгоритмов перебора по числу кластеров является использование жадного EM-алгоритма (см. [11, 12]), в котором компоненты смеси добавляются по одной, каждый раз решая задачу оптимизации для двухкомпонентной смеси. Однако, при этом значение результирующего правдоподобия, вообще говоря, может оказаться существенно ниже, чем при многомерной оптимизации. Кроме того, как показывают проведенные эксперименты по показателю Ранда и найденному числу кластеров, последовательные методы уступают по качеству алгоритму ARD EM.

Таким образом, в данной работе предложен метод ARD EM автоматического выбора числа кластеров в EM-алгоритме для разделения смесей нормальных распределений. Результаты экспериментов на модельных задачах показывают, что предложенный метод работает быстрее и точнее по сравнению с аналогами. Однако, алгоритм ARD EM имеет смысл применять лишь в тех случаях, когда сам EM-алгоритм разделения смесей нормальных распределений является адекватным методом кластеризации для рассматриваемой задачи. Известно (см. [13]), что это бывает далеко не всегда так. В последнем случае одним из возможных выходов является рассмотрение ансамблей методов кластеризации (см. [13, 14]).

Список литературы

1. *Dempster A. P., Laird N. M., Rubin D. B.* Maximum likelihood from incomplete data via the EM algorithm // J. Roy. Stat. Soc. B 1977. V. 39. P. 1–38.

2. *Bishop C. M.* Pattern Recognition and Machine Learning. New York: Springer, 2006.

3. *Tipping M. E.* Sparse Bayesian learning and the relevance vector machine // J. Mach. Learn. Res.. 2001. V. 1. P. 211-–244.

4. *MacKay D. J. C.* Bayesian interpolation // Neural Comp.. 1992. V. 4. No. 3. P. 415–447.

5. *Kyrgyzov I. O., Kyrgyzov O. O., Maitre H. ,Campedel M.* Kernel MDL to determine the number of clusters // Proc. Intern. Conf. Mach. Learn. Data Mining. Leipzig, Germany. 2007.

6. *Xu L., Jordan M. I.* On Convergence Properties of the EM Algorithm for Gaussian Mixtures // Neural Comp.. 1996. V. 8. P. 129–151.

7. *Rissanen J.* Modeling by shortest data description // Automatica. 1978. V. 14. P. 465–471.

8. *Freund Y., Schapire R. E.* A decision-theoretic generalization of on-line learning and an application to boosting // J. Comp. Syst. Sci.. 1997. V. 55. No. 1. P. 119–139.

9. *Akaike H. A.* A new look at the statistical model identification // IEEE Trans. Autom. Contr.. 1974. V. 19. No. 6. P. 716–723.

10. *Hubert L., Arabie P.* Comparing Partitions // J. Clas.. 1985. V. 2. P. 193–218.

11. *Vlassis N., Likas A.* A greedy EM algorithm for Gaussian mixture learning // Neural Processing Letters. 2000. P. 77–87.

12. *Verbeek J. J., Vlassis N., Krose B.* Efficient Greedy Learning of Gaussian Mixture Models // Neural Computation. 2003.

13. *Kuncheva L. I., Vetrov D. P.* Evaluation of Stability of k-Means Cluster Ensembles with Respect to Random Initialization // IEEE Trans. Pattern Anal. Mach. Intell.. 2005. V. 28. No. 11. P. 1798–1808.

14. *Рязанов В. В.* О синтезе классифицирующих алгоритмов на конечных множествах алгоритмов классификации (таксономии) // Ж. вычисл. матем. и матем. физ.. 1982. Т. 22. № 2. С. 429–440.