Министерство образования и науки Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)
Факультет информатики
Кафедра прикладной информатики

УДК 519.237.8

ДОПУСТИТЬ К ЗАЩИТЕ В ГАК

Зав. кафедрой, профессор, д.т.н.

_____С.П. Сущенко

«__»_____2015 г.

БАКАЛАВРСКАЯ РАБОТА

РАЗРАБОТКА АДАПТИВНЫХ БАЙЕСОВСКИХ КЛАССИФИКАТОРОВ И ИССЛЕДОВАНИЕ ИХ ЭФФЕКТИВНОСТИ НА ПРИМЕРЕ ЗАДАЧ РАСПОЗНАВАНИЯ АВТОРСКОГО СТИЛЯ ТЕКСТОВ

по основной образовательной программе подготовки бакалавров

010500 – Математическое обеспечение и администрирование информационных систем

Михалёва Ксения Александровна

Руководитель ВКР, профессор, д.т.н.

_____В.В. Поддубный
подпись

Автор работы, студент группы 1412

_____К.А. Михалёва
подпись

Электронная версия бакалаврской работы помещена в электронную библиотеку.

Администратор электронной библиотеки факультета

_____Е.Н. Якунина
подпись

Томск – 2015

Реферат

Дипломная работа 40 с., 8 рис., 3 табл., 9 источников, 1 приложение.

БАЙЕСОВСКИЙ КЛАССИФИКАТОР, НАИВНЫЙ БАЙЕСОВСКИЙ КЛАССИФИКАТОР, АВТОРСКИЙ СТИЛЬ, ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ, ГЛАВНЫЕ КОМПОНЕНТЫ, F-МЕРА, MATLAB

Объект исследования – алгоритмы байесовской классификации.

Цель работы – адаптация алгоритмов байесовской классификации к эмпирическим функциям распределения обучающих выборок и их сравнительный анализ.

Метод исследования – аналитический и вычислительный эксперимент.

Результаты работы – изучены и построены адаптированные алгоритмы байесовской классификации. Данные алгоритмы реализованы программно. Проведён вычислительный эксперимент по классификации текстов русской художественной прозы 19 века, приведены результаты сравнения эффективности классификации алгоритмов по F-мере Ван Ризбергена, получаемой по модельным данным.

Область применения – классификация объектов (образов).

Содержание

Введение

В машинном обучении классификацию понимают как задачу определения класса для ранее не встречавшегося образца (объекта) на основе эмпирических данных, так называемых прецедентов, которые описывают исследуемые образцы и отражают присущие им свойства и закономерности. Существует зависимость между образцами и классами, но она неизвестна. Множество прецедентов, пар образец-класс, составляет обучающую выборку, по которой находится зависимость, то есть строится алгоритм, способный для любого образца выдать ответ, к какому классу тот принадлежит. Это пример обучения с учителем. Под учителем в данном случае понимается обучающая выборка.

Примерами таких моделей, основанных на машинном обучении, являются байесовские классификаторы. В работе рассмотрены наивный байесовский классификатор [1,2] и оптимальный байесовский классификатор [3].

В байесовских классификаторах используется критерий, минимизирующий вероятность принятия ошибочного решения, поэтому байесовские алгоритмы являются статистически оптимальными [1,4]. Однако для этого алгоритмы требуют в идеале полного знания многомерных функций распределения наблюдаемых признаков для каждого класса. Необходимость такого знания обусловлена использованием формулы Байеса, которая лежит в основе байесовских методов принятия решения.

При практическом применении байесовских алгоритмов, как правило, возникают проблемы следующего рода. Во-первых, исследователь не обладает таким полным знанием, а во-вторых, объем обучающей выборки ограничен, поэтому не предоставляется возможность эмпирическим путем получить информацию о многомерном законе распределения наблюдаемых данных.

Для решения (точнее, для предотвращения) вышеупомянутых проблем изобрели наивный байесовский классификатор, который предполагает полное отсутствие статистических связей между признаками. Таким образом, необходимость полного знания многомерных функций распределения сводится к необходимости знания маргинальных функций распределения наблюдаемых признаков для каждого из исследуемых классов.

Обычно при построении наивного байесовского классификатора исходят из предположения о нормальности маргинальных функций распределения наблюдаемых признаков. В общем случае наблюдаемые признаки не подчиняются этому распределению.

Таким образом, целью данной работы является адаптация алгоритмов байесовской классификации к эмпирическим распределениям и сравнение их эффективности. В качестве примера взяты тексты русской художественной прозы 19 века.

Можно выделить следующие задачи:

1. Построение алгоритма преобразования эмпирических данных к гауссовским на основе копула-функций.
2. Построение алгоритма генерации данных со статистическими свойствами обучающих выборок.
3. Программная реализация построенных алгоритмов в среде программирования Matlab.
4. Сравнение эффективности построенных алгоритмов классификации на примере задач распознавания авторского стиля текстов.

1 Байесовская классификация

В настоящее время статистические методы широко применяются для классификации текстов по признакам авторского, жанрового, гендерного и других стилей. Байесовская теория принятия решений составляет основу статистического подхода к задаче классификации объектов. Этот подход основан на предположении, что задача выбора решения сформулирована в терминах теории вероятностей и известны все представляющие интерес вероятностные величины. В основе байесовской классификации лежит правило Байеса.

1.1 Постановка задачи

Рассмотрим обучающую выборку из n объектов, каждый из которых принадлежит одному из K классов и характеризуется набором m числовых признаков $a_1, a_2, \dots a_m$. Пусть имеется n_k объектов k-ого класса, так что $N = \sum_{k=1}^{K} n_k$. Значение j-ого признака i-ого объекта из k-ого класса обозначим x_{ijk}. Тогда этот объект можно охарактеризовать вектором-строкой $x_{ik} = (x_{i1k}, \dots, x_{ijk}, \dots, x_{imk})$. Эту строку будем рассматривать как i-ю реализацию векторной случайной величины ξ_k, подчиняющейся распределению вероятностей с плотностью $p(x_1, \dots, x_m|k)$, своей для каждого класса k [3].

Пусть теперь наблюдается объект, для которого необходимо определить, к какому классу он относится. Объект характеризуется только набором m числовых признаков $x_1, \dots, x_m$.

1.2 Общая структура байесовского классификатора

В основе классификатора лежит следующее правило. Классификатор вычисляет апостериорную вероятность $P(k|x)$ каждого класса k, которому

может принадлежать испытуемый объект, и относит этот объект к апостериорно наиболее вероятному классу $\hat{k}$:

$$\hat{k} = \arg\max_{k} \ln P(k|x_1, \dots, x_m).$$

Апостериорная вероятность вычисляется по формула Байеса:

$$P(k|x_1, \dots, x_m) = P(k)p(x_1, \dots, x_m|k)/p(k),$$

где $P(k)$ – априорная вероятность того, что объект относится к k-ому классу, $p(k)$ и $p(x_1, \dots, x_m|k)$ - безусловная и условная многомерные плотности распределения вектора признаков, компоненты которого обычно статистически зависимы.

Таким образом, байесовский классификатор предполагает, что многомерная совместная плотность распределения признаков известна для всех классов.

Аналитическое представление многомерной плотности вероятности известно только для нормального распределения. Вместе с тем многомерная нормальная плотность распределения дает подходящую модель для одного важного случая, а именно когда значения векторов признаков x для данного класса k представляются непрерывнозначными, слегка искаженными версиями единственного типичного вектора, или вектора-прототипа, μ_k. Именно этого ожидают, когда классификатор выбирается так, чтобы выделять те признаки, которые, будучи различными для образов, принадлежащих различным классам, были бы, возможно, более схожи для образов из одного и того же класса [1].

Многомерная нормальная плотность распределения в общем виде представляется выражением

$$p(x) = \frac{1}{(2\pi)^{\frac{m}{2}} \det R^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^T R^{-1}(x-\mu)},$$

где μ – m-компонентный вектор среднего значения, R – ковариационная матрица размера $m \times m$, T – знак транспонирования.

Отметим, что если все недиагональные элементы равны нулю, то $p(x)$ сводится к произведению одномерных нормальных плотностей компонент вектора x.

Поэтому для многомерного нормального распределения удаётся выразить в аналитически замкнутой форме (с точностью до несущественных слагаемых) алгоритм байесовской классификации:

$$\hat{k} = \arg\max_k \left(\ln P(k) - \frac{1}{2} \ln \det R_k - \frac{1}{2} (x_k - \mu_k) R_k^{-1} (x_k - \mu_k)^T \right), \quad (1)$$

где μ_k – m-вектор-строка математических ожиданий значений признаков объектов класса k, R_k – $m \times m$-матрица ковариаций векторов признаков класса k.

Диагональные элементы матрицы образуют m-вектор D_k дисперсий признаков объектов класса k.

1.3 Наивный байесовский классификатор

В наивном байесовском классификаторе делается предположение о независимости признаков объекта. Если пренебречь статистическими связями между компонентами вектора признаков, тогда матрица R_k будет диагональной с вектором D_k на главной диагонали и классификатор (1) станет наивным байесовским классификатором.

Также предполагается, что маргинальная плотность распределения $p(x_j|k)$ любого признака является нормальной для любого класса.

Но на практике так бывает далеко не всегда, то есть наблюдаемые данные не подчиняются нормальному закону распределения (в общем случае

закон вообще неизвестен) и имеет место статистическая зависимость, поэтому область применения классификатора сужается.

1.4 Оптимальный байесовский классификатор

Так как оптимальный байесовский классификатор является модификацией наивного байесовского классификатора, то в качестве решающего правила также берётся формула (1).

Основная идея состоит в том, чтобы, максимально используя обучающую выборку и гауссову копула-функцию [3,5,6], обойти два «наивных» предположения. Модификация позволяет, во-первых, учесть статистические связи между наблюдаемыми признаками. И во-вторых, адаптировать классификатор к неизвестному действительному распределению путем приведения сглаженных маргинальных функций распределения признаков к нормальному виду. Другими словами, с помощью нелинейных гауссовых копула-функций негауссовы данные преобразуются в гауссовы, которые можно подавать на вход классификатору.

Рассмотрим алгоритм байесовской классификации с обучением, который состоит из двух этапов:

1. Этап обучения.

   Если векторы математических ожиданий и дисперсий и матрицы R_k неизвестны, то по обучающей выборке $x_{ik} = \left(x_{i1k}, \dots, x_{ijk}, \dots, x_{imk}\right), i = \overline{1, n_k}, k = \overline{1, K}$, для каждого класса k строятся эмпирические оценки математического ожидания

$$\hat{\mu}_{jk} = \frac{1}{n_k} \sum_{i=1}^{n_k} x_{ijk}$$

   и дисперсии:

$$\hat{\sigma}_{jk}^2 = \frac{1}{n_k - 1}\sum_{i=1}^{n_k}(x_{ijk} - \hat{\mu}_{jk})^2.$$

Затем находятся эмпирические оценки ковариационной матрицы для каждого класса k:

$$\hat{R}_k = \frac{1}{n_k - 1}\sum_{i=1}^{n_k}(x_{ik} - \hat{\mu}_k)^T (x_{ik} - \hat{\mu}_k), \quad (2)$$

где $\hat{\mu}_k = (\hat{\mu}_{1k}, \dots, \hat{\mu}_{mk})$ – вектор-строка эмпирических оценок математических ожиданий значений признаков, T - знак транспонирования.

По найденным оценкам предоставляется возможным обучить байесовский классификатор (1) распознавать классы объектов, заменив в нем неизвестные μ_k и R_k на их оценки.

При этом необходимо уделить внимание тому факту, что обучающих выборок каждого класса должно быть достаточно для построения невырожденных матриц $\hat{R}_k$. Если возникает проблема дефицита, то решением может стать либо использование наивного байесовского классификатора, либо использование метода главных компонент [8], который позволит от исходной системы признаков $a_1, a_2, \dots a_m$ перейти к системе меньшей размерности, выбрав небольшое число первых главных компонент. Новые признаки в совокупности будут некоррелированными, но внутри каждого класса будут в общем случае коррелированными.

Теперь в случае, если признаки наблюдаемых объектов не подчиняются нормальному закону распределения вероятностей, необходимо провести специальную нормализацию признаков. Используя гауссову копула-функцию $C(x) = \Phi\left(\Phi_1^{-1}\left(F_1(x_1)\right), \dots, \Phi_n^{-1}\left(F_n(x_n)\right)\right)$, где $F_i(x_i), i = 1, 2, \dots, n$ – маргинальные функции распределения негауссовых наблюдений x, $\Phi_i^{-1}(z_i)$ – функции, обратные маргинальным функциям

распределения гауссова вектора z, $\Phi(z)$ – многомерная нормальная функция распределения, можно построить гауссову копула-оценку многомерного негауссова распределения и адаптировать гауссов байесовский классификатор к негауссовым данным следующим образом.

По обучающей выборке $x_{ik} = (x_{i1k}, \dots, x_{ijk}, \dots, x_{imk}), i = \overline{1, n_k}, k = \overline{1, K}$, для каждого класса k каждого признака a_j строятся маргинальные эмпирические функции распределения $\widehat{F}_j(x_j|k)$:

$$\widehat{F}_j(x_j|k) = \begin{cases} 0, \ x \le x_{(1)}, \\ \frac{n}{n_k}, \ x_{(n)} < x \le x_{(n+1)}, \ n = \overline{1, n_k - 1}, \\ 1, \ x > x_{(n_k)}, \end{cases} \quad (3)$$

где $x_{(n)}$ – n-ая порядковая статистика вариационного ряда наблюдений j-ого признака объектов класса k, n – ранг соответствующего наблюдения.

Эмпирическая функция распределения $\widehat{F}_j(x_j|k)$ является состоятельной статистической оценкой неизвестной маргинальной функции распределения j-ой компоненты вектора признаков объектов класса k.

Для построения преобразователя негауссовых наблюдений к гауссовым, необходимо сгладить эмпирические функции распределения. Мажорируем сверху и снизу ступенчатую непрерывную слева эмпирическую функцию распределения $\widehat{F}_j(x_j|k)$ непрерывными кусочно-линейными функциями (ломаными линиями) с точками излома в угловых точках графика функции (3). Примем среднее арифметическое этих ломаных за сглаженную непрерывную эмпирическую оценку $\tilde{F}_j(x_j|k)$ неизвестной непрерывной маргинальной функции распределения.

Затем обучающая выборка преобразуется в нормально распределённую выборку $z_{ik} = \left(z_{i1k}, \dots, z_{ijk}, \dots, z_{imk}\right)$ с $\hat{\mu}_{jk}$ и $\hat{\sigma}_{jk}^2$ по формуле:

$$z_{ij} = \Phi_{jk}^{-1}\left(\tilde{F}_j\left(x_{ij}\middle|k\right)\right), j = \overline{1,m}, \quad (4)$$

где Φ_{jk}^{-1} - обратные маргинальные гауссовы функции распределения с найденными эмпирическими оценками математических ожиданий и дисперсий для каждого класса.

Таким образом, наблюдаемое значение x_{ij} j-ого признака испытуемого объекта и соответствующее ему значение z_{ij} нормально распределенной случайной величины ξ имеют одинаковые вероятности, то есть являются эквивалентными в вероятностном смысле.

Нормально распределенная выборка $z_{ik} = \left(z_{i1k}, \dots, z_{ijk}, \dots, z_{imk}\right)$ используется в (2) вместо x_{ik}.

2. Этап классификации.

Наблюдаемый вектор-строка признаков $(x_1, \dots, x_m)$ испытуемого объекта преобразуется по формуле (4) в нормально распределённые векторы $z_k = \left(z_{1k}, \dots, z_{jk}, \dots, z_{mk}\right)$, $k = \overline{1,K}$ (для каждой гипотезы k о классе), по которым вычисляется логарифм функции правдоподобия каждого класса:

$$\ln p(z_k|k) = -\frac{1}{2}\ln\det\hat{R}_k - \frac{1}{2}(z_k - \hat{\mu}_k)\hat{R}_k^{-1}(z_k - \hat{\mu}_k)^T. \quad (5)$$

В качестве класса, к которому принадлежит испытуемый объект, выбирается максимально правдоподобный класс $\hat{\mathrm{k}}$ в соответствии с решающим правилом (1) для эквивалентных признаков z.

Вышеописанный алгоритм позволяет построить обученный и адаптированный к эмпирическому распределению наблюдений байесовский классификатор, который является оптимальным.

2 Метод главных компонент

Как упоминалось ранее, при недостаточном объеме обучающей выборки оптимальный байесовский классификатор построить невозможно. Выходом из такой ситуации может стать уменьшение размерности признакового пространства до значения, допускающего возможность построения невырожденных ковариационных матриц.

Уменьшить размерность признакового пространства позволяет известный в математической статистике метод главных компонент [8].

Признаковое пространство векторов x размерности m смешанного ансамбля объектов, объединяющего все классы, преобразуется в новое признаковое пространство векторов y той же размерности с некоррелированными между собой компонентами, называемыми главными компонентами. Это достигается поворотом осей m-мерного признакового пространства векторов x таким образом, чтобы проекции $y_1, y_2, \dots, y_m$ этих векторов на новые оси (главные компоненты) были некоррелированными между собой и чтобы проекция на первую ось (первая главная компонента y_1) обладала максимальной дисперсией, проекция на вторую ось (вторая главная компонента y_2) – максимальной дисперсией среди оставшихся, и т.д.

Такой поворот осей обеспечивается унитарным линейным преобразованием:

$$y = xC,$$

где x и y – m-векторы-строки исходных и новых признаков (главных компонент), C – унитарная $m \times m$-матрица факторных нагрузок ($CTC = CCT = I$, то есть $CT = C^{-1}$). Столбцы C_j матрицы C выражают коэффициенты корреляции главных компонент с исходными признаками и являются собственными векторами ковариационной матрицы R вектора

исходных признаков объединённого ансамбля объектов. Они удовлетворяют уравнению

$$RC_j = \lambda_j C_j$$

и отвечают расположенным в порядке убывания неотрицательным собственным числам λ_j симметричной матрицы R. Собственные числа λ_j, $j = \overline{1,m}$ являются корнями характеристического уравнения

$$\det(R - \lambda I) = 0$$

и выражают дисперсии некоррелированных главных компонент.

При нормальном распределении вектора x вектор главных компонент y также имеет нормальное распределение, но вследствие некоррелированности компонент его плотность факторизуется, распадается на произведение маргинальных плотностей главных компонент в объединённом ансамбле объектов. Однако, внутри классов главные компоненты будут всё же статистически зависимыми.

Выбирая в качестве нового набора признаков только часть $n < m$ наиболее изменчивых первых главных компонент с суммарной долей дисперсии

$$\gamma_n = \sum_{j=1}^{n} \lambda_j \Big/ \sum_{j=1}^{m} \lambda_j \leq 1,$$

можно существенно уменьшить размерность признакового пространства, благодаря чему может хватить числа объектов обучающей выборки для оценки не только математических ожиданий, но и ковариационных матриц главных компонент для каждого класса. Это позволит настроить классификатор (1) на вектора-признаки, составленные из первых $n < m$ главных компонент.

3 Генератор контрольных выборок

Рассмотренные ранее алгоритмы классификации требуют некоторый начальный набор данных – обучающую выборку – для построения (обучения). Далее обученный классификатор желательно протестировать, чтобы узнать качество классификации, но для этого необходим ещё один набор данных – контрольная выборка, объем которой должен быть достаточно большим. На практике возможен дефицит или полное отсутствие тестовых данных. Как в случае задачи классификации текстов русской художественной прозы 19 век, которая представляет интерес в данной работе. Поэтому необходимо использовать имитационное моделирование контрольной выборки со статистическими свойствами обучающей выборки.

Для генерации контрольной выборки набора признаков (модельных «относительных частот» со статистическими характеристиками обучающих выборок) объекта каждого класса используется следующий алгоритм [6].

1. стандартным датчиком нормально распределённых случайных чисел сгенерировать m независимых случайных чисел $\{v_1, \dots, v_m\}$;
2. преобразовать вектор-столбец v с независимыми компонентами в вектор-столбец $z = (z_1, \dots, z_m)^T$ с коррелированными компонентами по формуле:

$$z = Av,$$

где A - нижняя треугольная матрица в разложении Холецкого

$$R = AA^T,$$

где R - корреляционная матрица Пирсона, соответствующая ранговой корреляционной матрице Спирмена R_s, вычисленной по обучающей выборке, T – знак транспонирования;

3. преобразовать покомпонентно гауссов вектор z с коррелированными компонентами по формуле:

$$y_j = \Phi(z_j), j = \overline{1,m},$$

где Φ – маргинальная гауссова функция распределения (с нулевым математическим ожиданием и единичной дисперсией),

в вектор y со значениями в $[0,1]^m$, распределение которого описывается гауссовой копулой:

$$C_{\text{норм}}(y_1, \dots, y_m) = \Phi\left(\Phi^{-1}(y_1), \dots, \Phi^{-1}(y_m)\right),$$

где Φ^{-1} – обратная маргинальная стандартная гауссова функция распределения

4. преобразовать покомпонентно вектор y в искомый вектор x со сглаженными маргинальными эмпирическими распределениями $\widetilde{F_n}(x_j), j = \overline{1,m}$, построенными по обучающей выборке, и ранговой корреляционной матрицей Спирмена R_s по формуле:

$$x_j = \widetilde{F_n^{-1}}(y_j), j = \overline{1,m},$$

где $\widetilde{F_n^{-1}}$ - обратная сглаженная маргинальная функция распределения, построенная по обучающей выборке.

Полученный вектор x можно рассматривать как набор m числовых признаков только что написанного произведения конкретного автора. Таким образом, в совокупности данные тексты могут быть взяты в качестве обучающей выборки.

Сгенерировав необходимое количество наборов числовых признаков с характеристиками текстов для каждого автора и предъявив их классификатору, получим достаточно надежную эмпирическую оценку качества классификации.

4 Измерение качества классификации

Подавая на вход классификатора контрольные тексты, не участвовавшие в обучении классификатора, но принадлежность которых к классам известна, можно оценить эффективность (качество) работы исследуемого классификатора.

Для оценки качества классификации рассмотренных алгоритмов воспользуемся F-мерой.

F-мера складывается из полноты и точности и вычисляется для каждого класса в отдельности, а затем, если нужна общая F-мера, берётся среднее значение по всем классам [7].

Полнота по классу:

$$r_i = \frac{T_i}{M_i}, i = \overline{1, K},$$

где T_i – число текстов, правильно приписанных к классу i, M_i – общее число текстов класса i, имеющихся в тестируемых данных, K – число классов.

Точность по классу:

$$p_i = \frac{T_i}{C_i}, i = \overline{1, K},$$

где C_i – общее число текстов, приписанных к классу i (если $C_i = 0$, то считается $p_i = 0$).

F-мера по классу:

$$F_i = \frac{2}{(\frac{1}{p_i} + \frac{1}{r_i})} = \frac{2p_i r_i}{p_i + r_i}.$$

Если $p_i + r_i = 0$, то $F_i = 0$. Значение F-меры колеблется от 0 до 1.

Общая F-мера:

$$F = \frac{1}{K}\sum_{i} F_i \, , i = \overline{1, K}.$$

5 Программная реализация алгоритмов

Для реализации вышеописанных алгоритмов использовалась среда Matlab.

Основной причиной выбора именно этой среды разработки является то, что язык, инструментарий и встроенные математические функции позволяют исследовать и проводить анализ различных математических структур быстрее, чем с использованием традиционных языков программирования, таких как C/C++ или Java. Также несомненным плюсом является возможность использования встроенных функций построения 2D и 3D графиков для визуализации обрабатываемой информации или результатов.

5.1 Входные данные

Входные данные, то есть обучающая выборка, хранятся в файле с расширением *.xls* в следующем виде (таблица 1).

Таблица 1. Структура входных данных

$Признак_1$	$Признак_2$	…	$Признак_m$	Название класса
$Абсолютное\ значение_{11}$	$Абсолютное\ значение_{12}$	…	$Абсолютное\ значение_{1m}$	$Класс_1$
…	…	…	…	…
$Абсолютное\ значение_{N1}$	$Абсолютное\ значение_{N2}$	…	$Абсолютное\ значение_{Nm}$	$Класс_K$

В качестве признаков взяты служебные слова русского языка. В качестве названий классов – фамилии и инициалы русских классиков 19 века.

5.2 Генератор контрольной выборки

Реализация алгоритма генерации данных со статистическими свойствами обучающих выборок хранится в файле *mygenerator.m*.

Функция *mygenerator* в качестве входных параметров принимает массив объектов, принадлежащих конкретному классу, и объем контрольной выборки. На выходе – контрольная выборка указанного объема для данного класса.

Проиллюстрируем возможности имитационного моделирования на примере одного писателя – Н.В. Гоголя. Тогда обучающая выборка будет представлять собой матрицу относительных частот размерности 13×9, в которой строки соответствуют произведениям, а столбцы наблюдаемым признакам.

На рисунке 1 представлены эмпирические функции распределения $F_n(x)$, полученные по обучающей выборке (тонкие синие линии) при $n = 13$ и путем имитационного моделирования (черные полужирные линии) при $n = 200$ для каждого признака $j = \overline{1,9}$.

Также с помощью критерия Колмогорова и Смирнова [9] проверены гипотезы H_0 о равенстве исходного (по обучающей выборке) и модельного (по алгоритму генерации) эмпирических распределений для каждого признака. На рисунке 1 для каждого признака приведены результаты работы критерия: P – достигнутый уровень значимости и H – принимаемая гипотеза ($H = 0$ соответствует гипотезе H_0). Видно, что для каждого из 9 признаков нет оснований отвергнуть нулевую гипотезу, так как достигнутый уровень значимости превышает пороговое значение $P > 0.05$.. Это означает, что сгенерированные признаки соответствуют маргинальным статистическим характеристикам обучающей выборки.

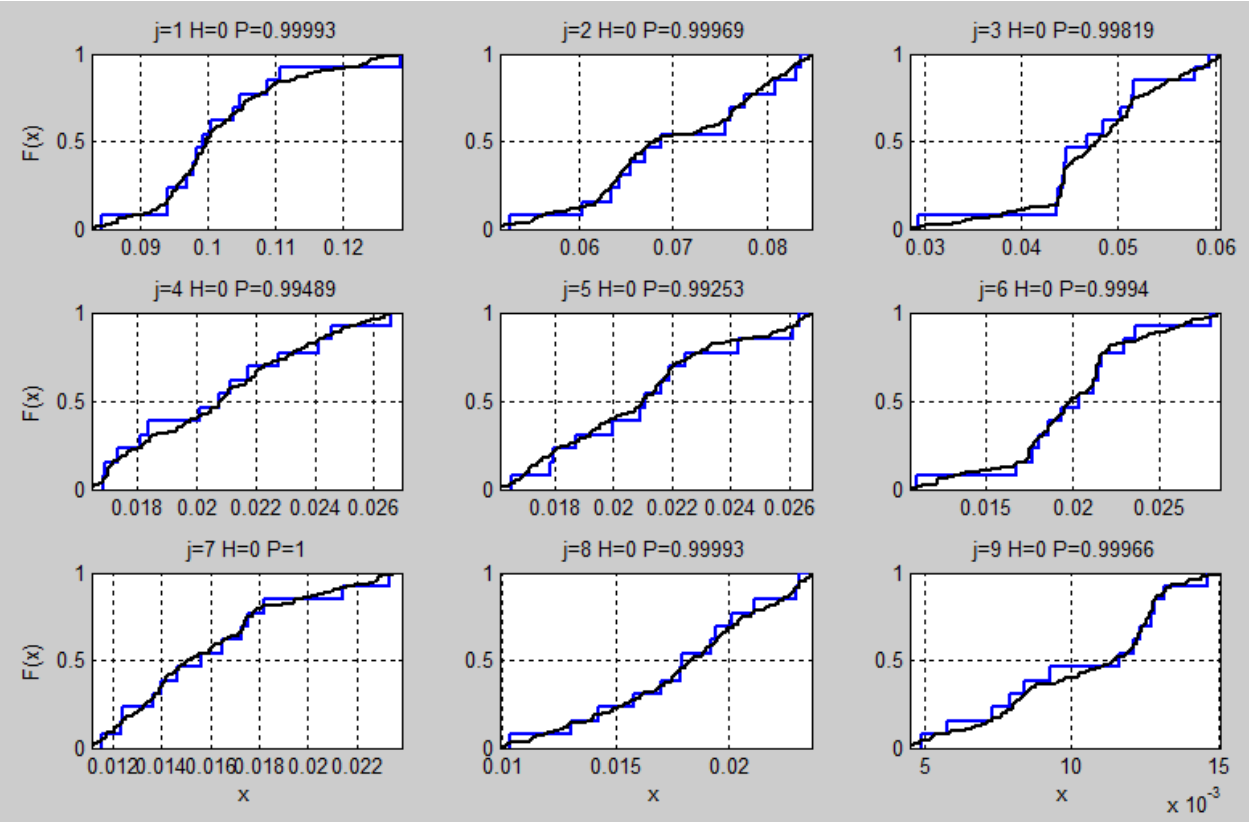

Рисунок 1 – Эмпирические функции распределения (ЭФР). Тонкие синие линии – исходные ЭФР, полужирные черные – модельные ЭФР

На рисунках 2 и 3 представлены столбиковые диаграммы эмпирических оценок математических ожиданий и стандартных отклонений для каждого признака $j = \overline{1,9}$ по обучающей выборке объема $n = 13$ и модельной выборке объема $n = 200$. Легко заметить, что математические ожидания практически совпадают, а различия стандартных отклонений пренебрежительно малы (самая большая разница наблюдается у первого признака, но даже она составляет всего ≈ 0.002).

На рисунке 4 показаны полутоновые яркостные изображения нормированных корреляционных матриц Пирсона и Спирмена построенных по обучающей выборке и модельной выборке. Белый цвет характеризует сильную корреляцию, черный – слабую корреляцию.

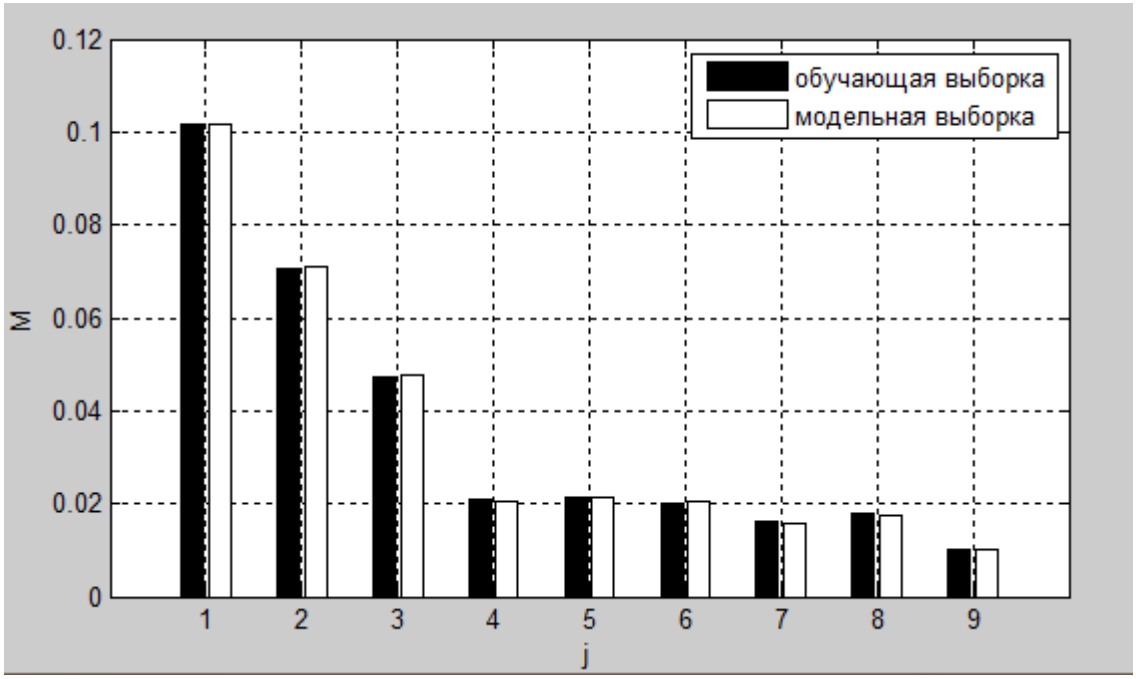

Рисунок 2 – Эмпирические оценки математических ожиданий по обучающей и модельной выборкам

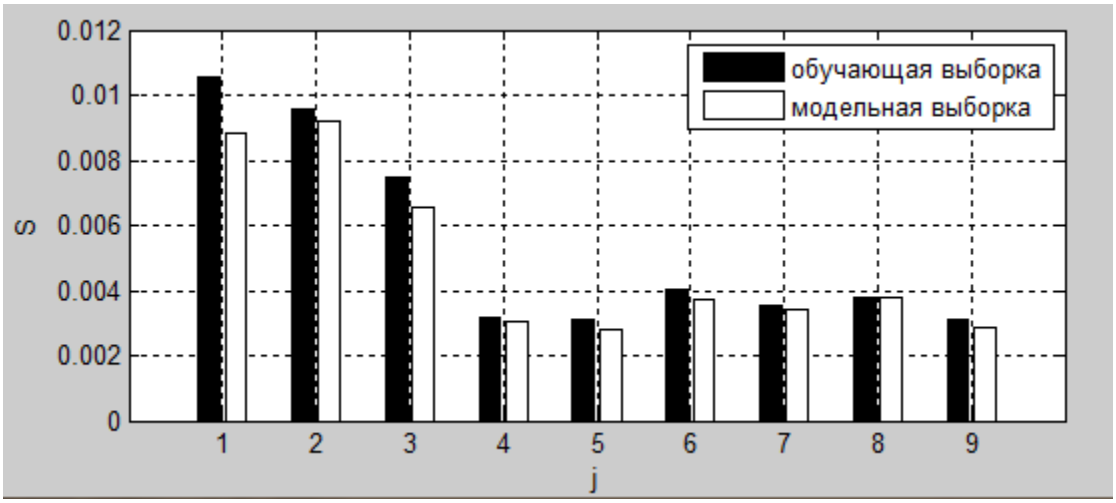

Рисунок 3 – Эмпирические оценки стандартных отклонений по обучающей и модельной выборкам

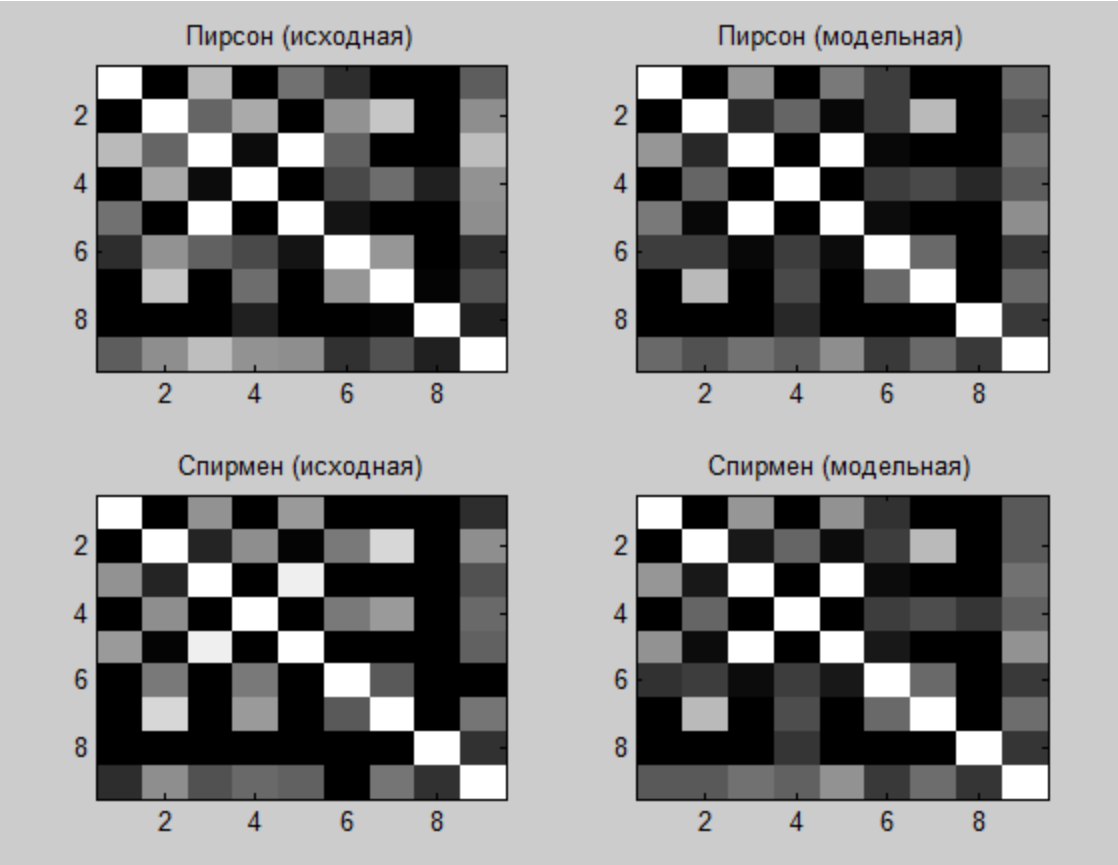

Рисунок 4 – Полутоновые яркостные изображения нормированных корреляционных матриц

5.3 Алгоритмы байесовской классификации

Реализация адаптированных алгоритмов байесовской классификации хранится в файле *newcopula.m* и состоит из этапа обучения и этапа классификации для вычисления F-меры.

При запуске *newcopula.m* сначала происходит считывание обучающей выборки из **.xls* файла. Считанные данные обрабатываются следующим образом. Абсолютные частоты преобразуются в относительные, далее относительные частоты преобразуются по методу главных компонент, который реализован в функции *princomp*, в «новые признаки», которые разбиваются по классам (вопрос, какое именно число главных компонент выбирается для построения классификатора, будет рассмотрен в следующем

разделе). Затем для каждого класса с использованием функции *matrixC* считаются оценки ковариационных матриц. На этом этап обучения заканчивается.

На этапе классификации, используя функцию *mygenerator*, генерируется необходимое число объектов контрольной выборки. Для каждого объекта контрольной выборки вызывается функция *logarithmK* в предположении принадлежности данного объекта к конкретному классу. Данная функция сначала по формуле (4) преобразует признаки объекта в нормально распределенные величины (рисунок 5). На рисунке 5 представлен пример графиков эмпирического, сглаженного эмпирического и нормального распределений с одинаковыми эмпирическими средними и дисперсиями для относительных частот употребления служебного слова «в» в 13 произведениях Н.В. Гоголя. Для любого значения относительной частоты *x* на оси абсцисс по графику $F_э(x)$ сглаженной эмпирической функции распределения можно найти её значение и по равной ординате графика $\Phi(z)$ нормальной функции распределения найти соответствующее значение нормально распределённой абсциссы z этого графика, эквивалентное в вероятностном смысле исходной относительной частоте *x*. На графике показано данное преобразование пунктирной линией, то есть проход по пути от точки *x* на оси абсцисс вертикально вверх до точки пересечения с кривой $F_э(x)$, затем горизонтально вправо до точки пересечения с кривой $\Phi(z)$, затем вертикально вниз до точки z пересечения с осью абсцисс.

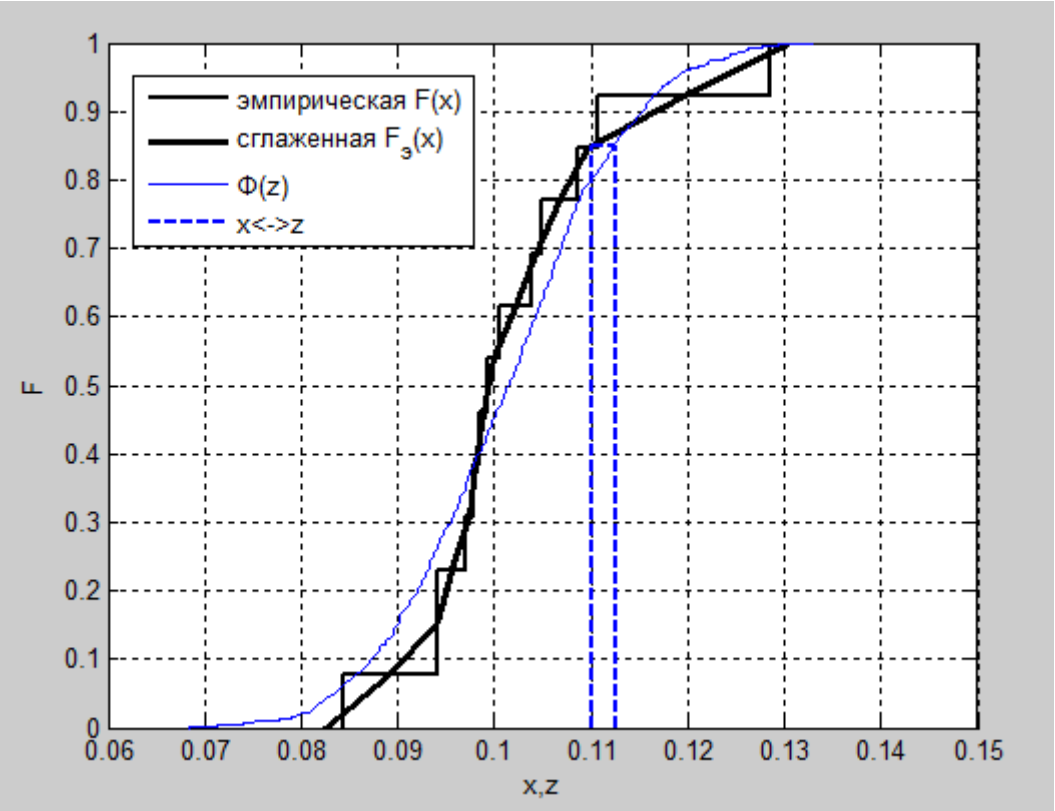

Рисунок 5 – Геометрическая интерпретация формулы (4)

Затем, используя полученные значения признаков z, функция *logarithmK* считает логарифм функции правдоподобия по формуле (5) (на выходе функции - значение логарифма правдоподобия для конкретного класса).

Описанная последовательность действий повторяется для каждого класса обучающей выборки. Затем в соответствии с решающим правилом (1) выбирается класс, к которому принадлежит объект. Этап классификации заканчивается подсчётом F-меры.

Так как вышеописанная последовательность действий присуща оптимальному байесовскому классификатору, то следует описать условие, при котором получится наивный байесовский классификатор. Если на этапе обучения после вычисления оценок ковариационных матриц вместо всей матрицы взять диагональную, то получится адаптированный наивный байесовский классификатор.

На рисунке 6 представлены эмпирические функции распределения, полученные по контрольной выборке после ее нормализации по формуле (4) (черные полужирные линии), и нормальная функция распределения с математическим ожиданием и дисперсией j-ого признака (в качестве примера – Н.В. Гоголь с 13 произведениями) (тонкие синие линии) для каждого признака $j = \overline{1,9}$.

Также на рисунке 6 представлены результаты исследования качества алгоритмов преобразования эмпирических данных к гауссовским. Проверка нулевой гипотезы состоящей в том, что выборка получена из генеральной совокупности имеющей нормальное распределение с заданными средним и средним квадратическим отклонением выполняется на основе статистики Z. На рисунке 6 для каждого признака приведены результаты исследования: P – достигнутый уровень значимости и H – принимаемая гипотеза ($H = 0$ соответствует гипотезе H_0). Видно, что для каждого из 9 признаков нет оснований отвергнуть нулевую гипотезу, так как достигнутый уровень значимости превышает пороговое значение $P > 0.05$.

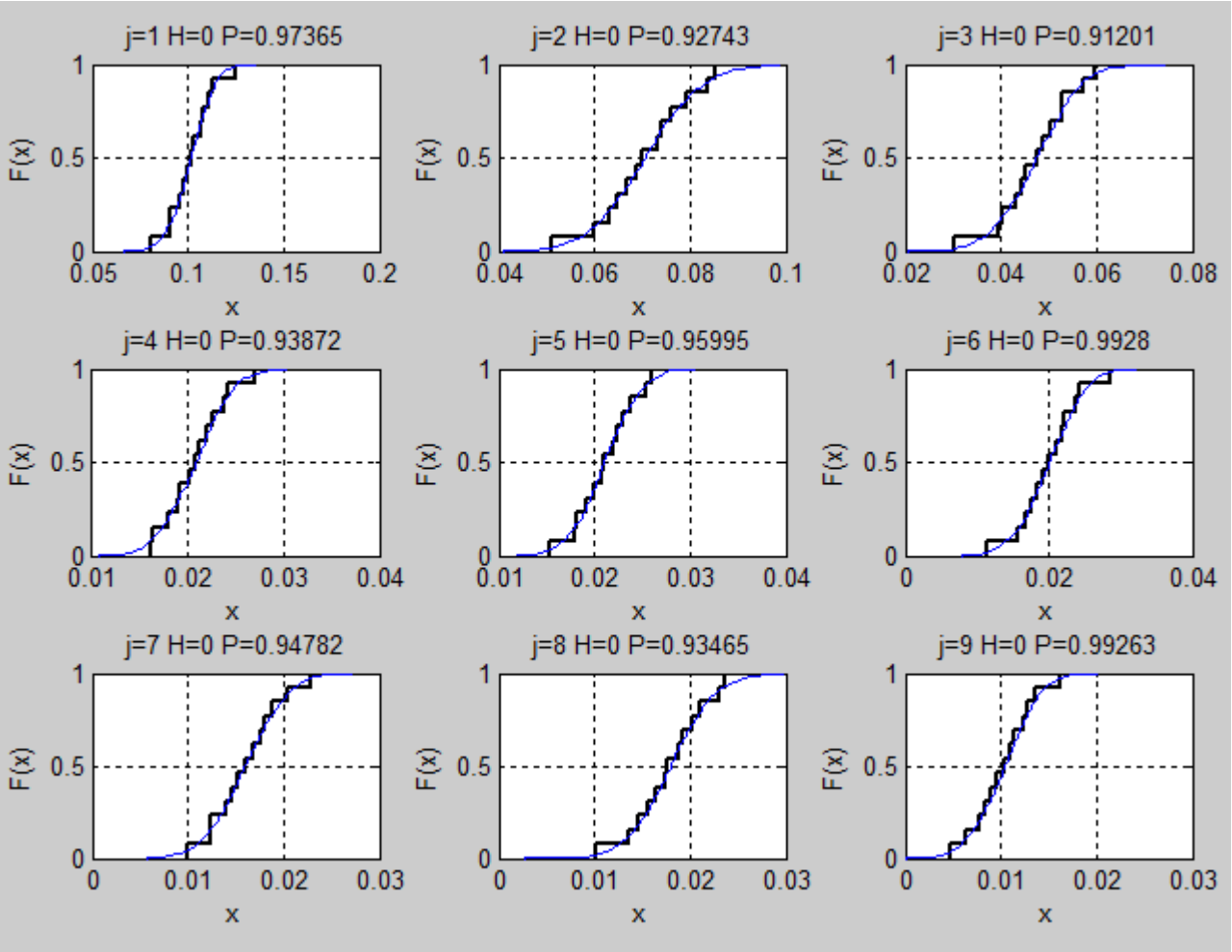

Рисунок 6 – Тонкие синие линии – нормальные функции распределения, полужирные черные линии – ЭФР признаков z

6 Сравнение эффективности алгоритмов классификации

Сравнительное исследование эффективности классификации текстов адаптированными наивным и модифицированным байесовскими классификаторами в условиях дефицита или отсутствия контрольных текстов, то есть контрольных выборок, проводится с помощью имитационного моделирования тестового признакового пространства (алгоритма генерации данных со статистическими свойствами обучающих выборок) с предъявлением их исследуемым классификаторам и с последующим подсчётом *F*-меры, усреднённой по всем классам.

6.1 Подготовка обучающих данных

Сначала в качестве обучающей выборки (таблица 2) предполагалось использовать 155 текстов художественных произведений 12 авторов с количеством признаков равным 53, точнее говоря, относительные частоты этих признаков.

Относительные частоты (доли абсолютных частот употребления того или иного служебного слова в общем числе употребления в тексте служебных слов) получаются делением содержимого ячеек каждой строки на сумму содержимого всех ячеек этой строки. Это позволяет избежать искажений результатов классификации из-за различий в объёмах текстов, непосредственно влияющих на значения абсолютных частот.

В результате каждому тексту ставится в соответствие вектор-строка, содержащая в качестве признаков значения относительных частот. Набор таких числовых характеристик для каждого текста образует вектор признаков стиля этого текста.

Таблица 2. Структура обучающей выборки

Автор	Количество произведений
Гоголь	13
Гончаров	3
Достоевский	26
Куприн	2
Лермонтов	5
Лесков	7
Пушкин	10
Салтыков-Щедрин	5
Толстой	8
Тургенев	40
Чернышевский	1
Чехов	35

Однако обучающая выборка такого объема не позволила построить невырожденные ковариационные матрицы для каждого класса, поэтому было принято решение о проведении специальной подготовки обучающих данных, направленной на увеличение числа обучающих объектов с сохранением их представительности. Подготовка состоит в том, что исходные тексты обучающей выборки разбиваются на блоки (по 10-20Кб каждый).

Чтобы убедиться, что разбиение произведений на блоки не приведет к потере статистических свойств класса, возьмем в качестве примера автора – Н.В. Гоголя. На рисунке 7 представлены эмпирические функции распределения, полученные по исходной обучающей выборке объема $n = 13$

(полужирные синие линии) и подготовленной обучающей выборке объема $n = 42$ (тонкие черные линии) для каждого признака $j = \overline{1,9}$. С помощью критерия Колмогорова и Смирнова проверены гипотезы H_0 о равенстве данных эмпирических распределений для каждого признака. На рисунке 1 для всех признаков приведены результаты работы критерия: P – достигнутый уровень значимости и H – принимаемая гипотеза ($H = 0$ соответствует гипотезе H_0). Видно, что для каждого из 9 признаков нет оснований отвергнуть нулевую гипотезу, так как достигнутый уровень значимости превышает пороговое значение $P > 0.05$.. Это означает, что специальная подготовка обучающих данных сохраняет представительность обучающих объектов.

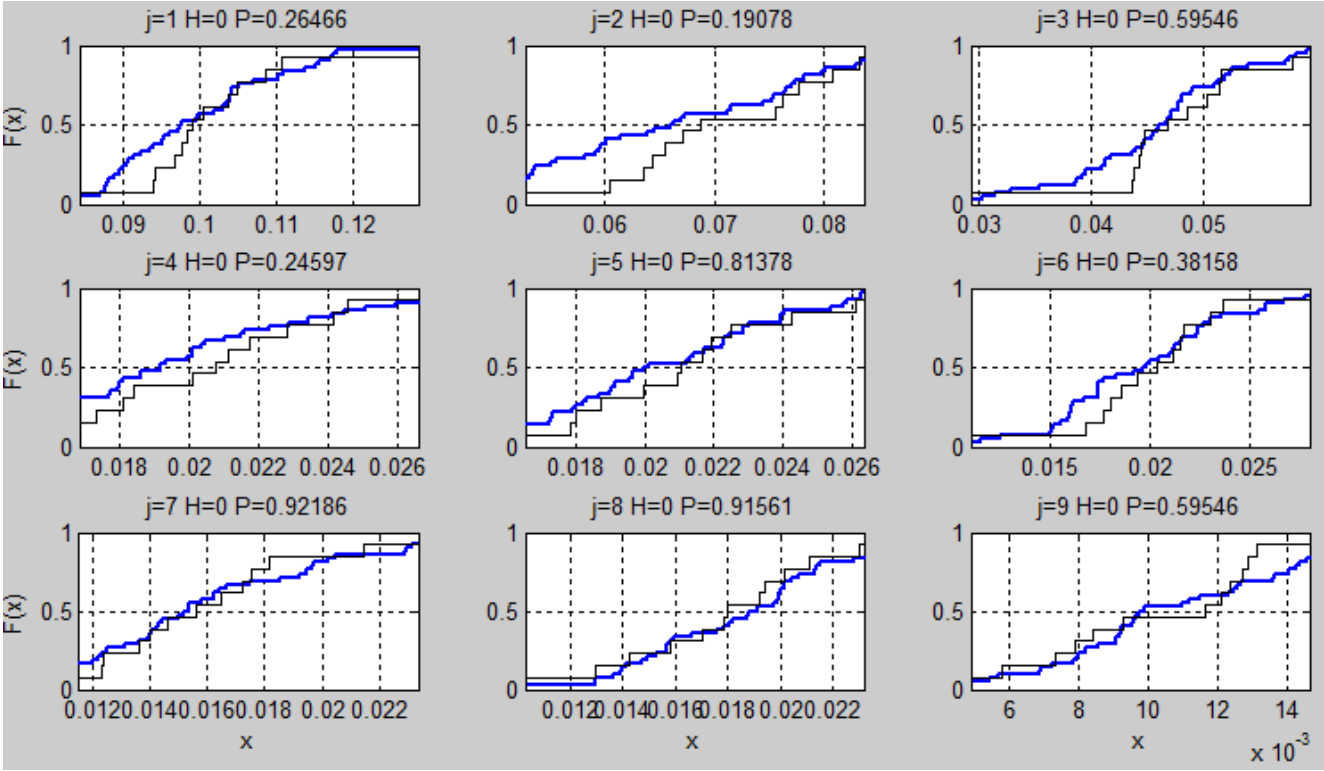

Рисунок 7 – Эмпирические функции распределения (ЭФР). Полужирные синие линии – ЭФР по исходной обучающей выборке, тонкие черные – ЭФР по подготовленной обучающей выборке

В результате такой подготовки обучающая выборка увеличилась до 772 блоков текстов.

К сожалению, при количестве признаков равном 53 даже такого объема обучающей выборки недостаточно для эмпирического оценивания параметров распределения, поэтому необходимо перейти к новым признакам – главным компонентам.

На рисунке 8 представлена «каменистая осыпь» дисперсий главных компонент набора признаков.

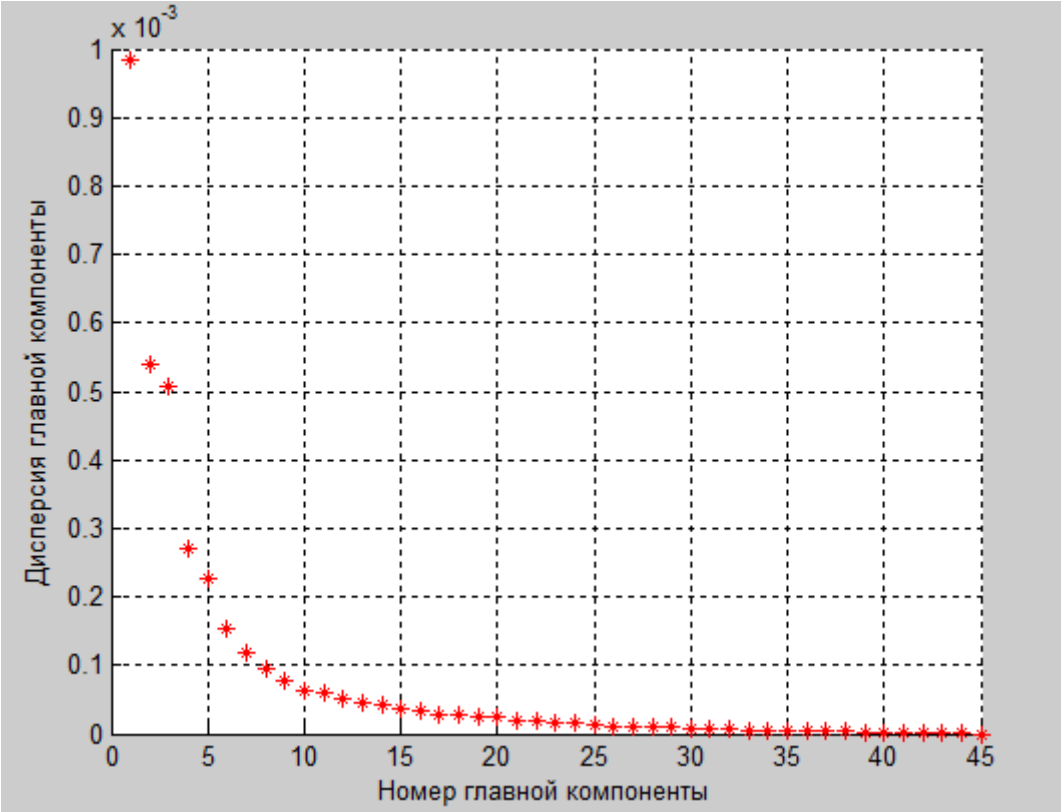

Рисунок 8 – «Каменистая осыпь» дисперсий главных компонент

Результаты сравнительного анализа по F-мере качества классификации текстов по главным компонентам байесовскими классификаторами представлены в таблице 3. Контрольная выборка со статистическими свойствами обучающей выборки получена с помощью имитационного моделирования в объеме $n = 100$ для каждого класса.

Таблица 3. Результаты тестирования алгоритмов

Кол-во главных компонент	% от общей дисперсии	Оптимальный байесовский классификатор	«Наивный» байесовский классификатор
1	31	0,16	0,17
2	44	0,32	0,27
3	56	0,45	0,46
4	63	0,56	0,50
5	69	0,64	0,56
6	73	0,71	0,67
7	76	0,80	0,74
8	78	0,87	0,79
9	80	0,90	0,82
10	82	0,92	0,83
11	84	0,93	0,85
12	85	0,93	0,85

По результатам, представленным в таблице 3, можно сделать вывод, что при достаточном количестве главных компонент оптимальный байесовский классификатор обеспечивает более высокое качество классификации, чем «наивный».

Заключение

В ходе работы исследованы алгоритмы байесовской классификации. Также изучены и исследованы алгоритмы генерации данных со статистическими свойствами обучающих выборок и преобразования эмпирических данных к гауссовским на основе копула-функций. Все вышеперечисленные алгоритмы реализованы программно.

Основная цель исследования состояла в том, чтобы поставить алгоритмы в одинаковые условия, адаптировать под обучающую выборку, и принять решение, стоит ли модифицировать наивный байесовский классификатор, то есть учитывать статистические зависимости между признаками, или модификация нерациональна, и достаточно использовать наивный байесовский классификатор.

Основываясь на результатах, полученных в исследовании, можно прийти к заключению, что модификация целесообразна.

Список использованной литературы

1. Дуда Р., Харт П. Распознавание образов и анализ сцен. / Пер. с англ. – М.: Мир, 1976. – 511 с.
2. Наивный байесовский классификатор. – [Электронный ресурс]. – URL: http://bazhenov.me/blog/2012/06/11/naive-bayes.html (Дата обращения 26.05.2015).
3. Кубарев А.И., Поддубный В.В. Байесовская классификация с обучением на основе использования копула-функций // Информационные технологии и математическое моделирование (ИТММ-2013): Материалы XII Всероссийской научно-практической конференции с международным участием им. А.Ф. Терпугова (29-30 ноября 2013г.). – Томск: Изд-во Том. ун-та, 2013. – Ч.2. – С.126–130.
4. Прикладная статистика: Классификация и снижение размерности: Справочное издание / С.А. Айвазян, В.М. Бухштабер, И.С. Енюков, Л.Д. Мешалкин / Под ред. С.А. Айвазяна. – М.: Финансы и статистика, 1989. – 608 с.
5. Кубарев, А. И., Поддубный, В. В. Адаптивная байесовская классификация объектов в метрическом пространстве. // Новые информационные технологии в исследовании сложных структур: Материалы Десятой российской конференции с международным участием. – Томск: Изд. Дом Том. гос. ун-та, 2014. – С. 115–116.
6. Поддубный В.В., Пехтерев А.С. Копулы сглаженных эмпирических распределений при наличии связок (совпадений) и их применение в имитационном моделировании // Труды XII Международной ФАМЭБ’2013 конференции. / Под ред. Олега Воробьёва. – Красноярск: НИИППБ, СФУ, 2013. – С. 312–321.

7. Шевелев О.Г. Методы автоматической классификации текстов на естественном языке: Учебное пособие. Томск: ТМЛ-Пресс, 2007. – 144с.
8. Афифи А., Эйзен С. Статистический анализ: Подход с использованием ЭВМ. / Пер. с англ. – М.: Мир, 1982. – 488 с.
9. Закс Л. Статистическое оценивание. / Пер. с нем. В.Н. Варыгина. / Под ред. Ю.П. Адлера, В.Г. Горского. – М.: Статистика, 1976. – 598 с. с ил.

Приложение А. Руководство программиста

Следует рассказать подробнее о реализации исследуемых алгоритмов.

newcopula.m – файл программа (Script M-File), которая отвечает за специальную подготовку данных, также за построение (обучение) и тестирование байесовских классификаторов. Так как подробное описание функции изложено в главе 5.3, уделим внимание основным моментам, которые не были упомянуты ранее. Для изменения обучающей выборки необходимо поменять значение переменной filename, в которой хранится путь до файла **.xls* с обучающей выборкой в формате таблицы 1. Как упоминалось ранее, объема обучающей выборки недостаточно для построения невырожденных ковариационных матриц, поэтому от обучающей выборки (матрицы относительных частот, которая хранится в переменной *M1*) осуществляется переход к главным компонентам (матрица score, также требуется указать до какого количества признаков уменьшить признаковое пространство, то есть задать необходимое количество главных компонент). В зависимости от того, какую матрицу использовать на этапе обучения классификатора: матрицу ковариаций (вызов функции *matrixC*) или диагональную матрицу дисперсий (в таком случае необходимо привести ковариационную матрицу к диагональной путем двойного вызова функции *diag*). Для изменения объема контрольной выборки требуется поменять значение переменной *samplecount*. После выполнения в командном окне (Command Window) отобразится значение качества классификации, то есть значение F-меры.

Используемые файлы функции (Function M-Files) описаны в нижеприведенном списке.

1. *trnsit.m* – позволяет транслитерировать русское название класса в латиницу.

2. *expanddisp.m* – на вход – матрица одного класса, на выходе: вектор-строка математических ожиданий и вектор-строка среднеквадратичных отклонений для каждого признака.

3. *funcemprdistr.m* – на вход: вектор-столбец (один признак одного класса), второй параметр – если 1, то эмпирическая функция распределения, любое другое число – обратная ЭФР, третий параметр – в соответствии второму: либо значение признака, либо значение вероятности. На выходе – либо вероятность для переданного значения признака, либо значение признака для переданной вероятности.

4. *logarithmK.m* – на вход – матрица одного класса, вектор-строка признаков (один объект одного класса) из контрольной выборки и матрицу ковариаций, соответствующая классу, переданному в качестве первого параметра. На выходе – значение логарифма правдоподобия. Далее приведен полный код функции (Пример кода 1). В цикле по *jj* значения вектора-строки преобразуются в нормальные величины, по которым высчитывается логарифм правдоподобия.

```
function [ l ] = logarithmK( X, x1,Rk )

[~,m]=size(X);
[mu3,sigma3]=expanddisp(X);
for jj=1:1:m
        p=funcemprdistr(X(:,jj),1,x1(1,jj));
        z1(1,jj)=norminv(p,mu3(1,jj),sigma3(1,jj));
end;
l=-log(det(Rk))/2-(z1-mu3)*pinv(Rk)*(z1-mu3)'/2;
if sum(abs(z1-mu3))==Inf
   l=-Inf;
end
end
```

Пример кода 1 – Функция logarithmK

5. *matrixC.m* – на вход – матрица одного класса, на выходе – оценка матрицы ковариаций для данного класса. Полный код

функции приведен ниже (Пример кода 2). В двойном цикле матрица класса преобразуется поэлементно в матрицу нормально распределенных величин, по которым в следующем цикле находится оценка ковариационной матрицы.

```
function [ Rk ] = matrixC( X )

[n,m]=size(X);
[mu3,sigma3]=expanddisp(X);
Rk=zeros(m,m);
z=[];
for ii=1:1:n
    for jj=1:1:m
        p=funcemprdistr(X(:,jj),1,X(ii,jj));
        z(ii,jj)=norminv(p,mu3(1,jj),sigma3(1,jj));
    end;
end;
for ii=1:1:n
    Rk=Rk+(z(ii,:)-mu3)'*(z(ii,:)-mu3);
end;
Rk=Rk/(n-1);
end
```

Пример кода 2 – Функция matrixC

6. *mygenerator.m* – на вход: первый параметр – матрица одного класса, второй параметр – количество объектов, которые будут получены путем имитационного моделирования, на выходе – матрица смоделированных признаков, то есть совокупность объектов данного класса. Алгоритм описан в разделе 3, поэтому ограничимся приведением полного кода функции (Пример кода 3).

```
function [ xj,R ] = mygenerator( MATR, kol) %на вход под
%на выходе матрица смоделированных признаков(каждая стро
[~,m]=size(MATR);
R=matrixC(MATR);
A=chol(R)';
xj=[];
for iii=1:1:kol
v=randn(m,1);
z=A*v;
for jj=1:1:m
    yj(jj,1)=normcdf(z(jj,1),0,sqrt(R(jj,jj)));
end;
for jj=1:1:m
    xjj(jj,1)=funcemprdistr((MATR(:,jj))',0,yj(jj,1));
end;
xj(iii,:)=xjj';
end;
end
```

Пример кода 3 – Функция mygenerator