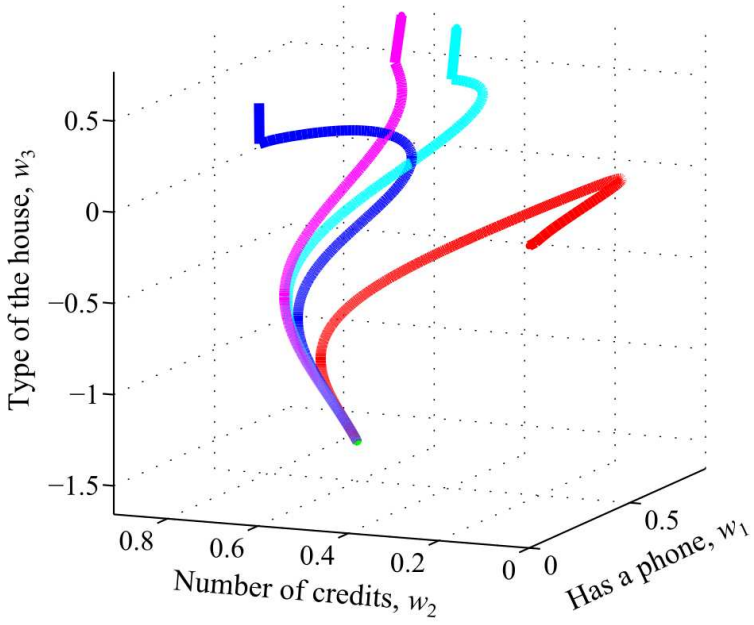
ISSN 2223-3792

Машинное обучение и анализ данных

Май 2012 **Том 1, номер 3**



Машинное обучение и анализ данных

Journal of Machine Learning and Data Analysis
ISSN 2223-3792 Rus

Цель журнала — развитие теории машинного обучения и интеллектуального анализа данных и методов проведения вычислительных экспериментов. Приветствуются работы студентов и аспирантов, а также обзорные, фундаментальные и методические статьи исследователей, работающих в области машинного обучения.

Тематика журнала:
— регрессионный анализ,
— классификация,
— кластеризация,
— многомерный статистический анализ,
— байесовские методы регрессии и классификации,
— методы прогнозирования временных рядов,
— методы оптимизации в задачах машинного обучения и анализа данных,
— методы визуализации данных,
— обработка и распознавание речи и изображений,
— анализ и понимание текста, информационный поиск,
— прикладные задачи анализа данных.

Вычислительный центр Российской академии наук
Московский физико-технический институт
Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»

Москва, 2012

Содержание

Сравнение формы бинарных растровых изображений на основе скелетизации*

Кушнир О. А.
kushnir-olesya@rambler.ru
Тула, Тульский государственный университет

Данная работа посвящена проблеме сравнения формы бинарных растровых изображений на основе скелетных графов. Проводится анализ существующих подходов к сравнению скелетных графов, заключающихся в применении к этим графам различных методов классификации на основе векторов признаков, мер, метрик. Также ставится задача нахождения метрики, заданной в пространстве скелетных графов, которая позволила бы эффективно сравнивать формы произвольных объектов в реальном времени путем применения универсального классификатора, построенного на методе опорных векторов.

Ключевые слова: форма бинарного растрового изображения, скелетный граф, метрика, потенциальная функция.

Введение

Задача распознавания формы объектов, представляемых в виде бинарных изображений, возникает во многих приложениях, например, при биометрической идентификации личности, распознавании позы и жестов человека, в системах оптического распознавания символов и пр. Наиболее удачной моделью формы является описание в виде скелетного графа, представляющего собой срединные оси фигуры. Существуют различные методы построения этого графа (они описаны в разделе 1 данной работы), и на основе результатов скелетизации фигуры различными методами можно выделять различные признаки для решения задачи классификации формы фигур.

В настоящее время не существует универсального метода для сравнения форм бинарных растровых изображений. Почти все имеющиеся меры сходства строятся на основе априорной информации о классифицируемых объектах, исходя из специфики конкретной прикладной задачи. Следовательно, для каждого нового приложения вопрос определения меры сходства решается заново. В качестве универсальной метрики в пространстве графов рассматривается только редакционное расстояние (edit distance) между двумя скелетными графами. Но при построении данной метрики для графов с большим количеством вершин возникает проблема большой вычислительной сложности, решаемая только приближенными методами (см. раздел 2 данной работы).

Следовательно, существует открытая задача нахождения метрики в пространстве скелетных графов. Причем необходим эффективный алгоритм ее вычисления, позволяющий сравнивать объекты в реальном времени. Последнее требование предъявляется исходя из практических приложений распознавания образов.

Наиболее эффективным на сегодняшний день методом классификации данных является метод опорных векторов (Support Vector Machines – SVM). Он является линейным классификатором, но для применения его на линейно неразделимых выборках используется переход от используемых в SVM скалярных произведений к произвольным потенциальным функциям. Потенциальные функции могут быть построены, опираясь на метрики специального вида (так называемые евклидовы метрики [8]).

Научный руководитель О. С. Середин

В третьей части данной работы дан краткий обзор теоретических положений метода опорных векторов на основе потенциальных функций и приведен пример потенциальной функции, построенной на метрике редакционного расстояния между графами.

В качестве цели для дальнейших исследований ставится задача нахождения приемлемой метрики и на ее основе – потенциальной функции, которая позволила бы эффективно решать задачу сравнения форм произвольных объектов в реальном времени путем применения универсального классификатора для форм бинарных растровых изображений, оставаясь в рамках теории линейных методов анализа эмпирических данных на попарных методах сравнения объектов.

1. Скелетизация бинарных растровых изображений для задач описания формы

Если аппроксимировать черные точки бинарного растрового изображения непрерывной замкнутой областью, то *скелетом* замкнутой области называется геометрическое место точек области, являющихся центрами максимальных по включению вписанных окружностей [5] (см. рис. 1).

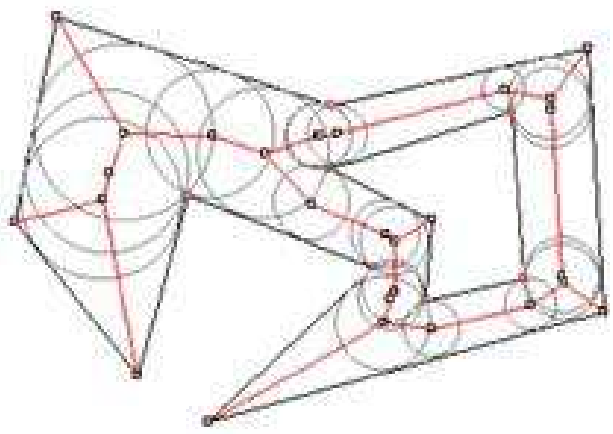


Рис. 1. Скелет замкнутой области

Скелет представляет собой плоский граф, структура которого очень хорошо отражает особенности формы объекта на изображении. Вследствие этого скелетизация широко используется для решения задач анализа и распознавания формы объектов в системах компьютерного зрения.

Для того чтобы распространить понятие скелета на дискретные растровые изображения, применяются два основных принципиально различных подхода, которые условно можно назвать дискретным и непрерывным.

Дискретный подход к скелетизации использует определение скелета, основанное на метафоре «пожара в прерии». Предполагается, что по границе области одновременно вспыхивает огонь, который распространяется внутри нее по всем направлениям с постоянной скоростью. Те точки области, в которых сходятся два или более огненных фронта, являются по определению точками скелета. Точка схода фронтов равноудалена от ближайших точек возгорания на границе. «Пожар в прерии» служит основой для определения и построения скелета в терминах растровых бинарных изображений. Подмножество черных точек растра рассматривается как дискретный образ некоторой замкнутой области, граничные точки растрового пятна – как образ границы области, а непрерывное распростра-

нение огня моделируется дискретным процессом последовательного «сжигания» соседних черных точек растра.

Процесс построения скелета дискретного образа реализуется в различных алгоритмах «утончения» [4, 5], в алгоритме «дистанционных карт» [5], в «волновом» алгоритме [9]. Последовательное «сжигание» соседних черных точек растра на примере метода топологического утончения приведен на рис. 2.

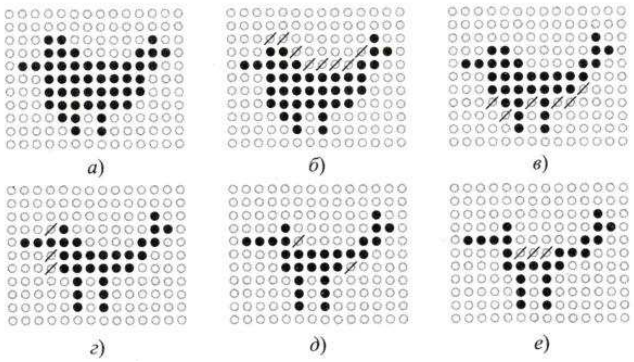


Рис. 2. Метод топологического утончения

В результате построения скелета для дискретного бинарного изображения получается новое дискретное бинарное изображение.

Следует отметить, что дискретные скелеты обладают следующими недостатками. Во-первых, расстояние между точками растра при их построении измеряется не в евклидовой метрике, а в зависимости от используемого понятия 4- или 8-соседства точек растра, в метрике l_1 («манхэттенской») или l_∞(«шахматной»). Это значит, что в качестве вписанной в область окружности выступает квадрат. Неевклидовы метрики приводят к неустойчивости получаемых скелетов по отношению к простым преобразованиям исходного изображения.

Во-вторых, дискретный скелет вычисляется неоднозначно и зависит от последовательности анализа граничных точек образа. Успешное построение дискретных аналогов скелета возможно, как правило, если дискретные фигуры имеют простую структуру.

Непрерывный подход к построению скелета растрового образа развит и хорошо исследован в работах школы Л.М. Местецкого [5, 6, 7, 11, 13] и состоит в аппроксимации границы дискретной фигуры многоугольником минимального периметра и построении скелета области, ограниченной этим многоугольником, в соответствии с формальным его определением, что скелет – это множество тех точек фигуры, для которых существует не менее двух равноудаленных ближайших точек границы области. Итогом применения данного подхода к растровому изображению является математическое описание скелета аппроксимирующей данное изображение «непрерывной» замкнутой области.

Скелет чрезвычайно чувствителен к локальным свойствам границы образа. С каждой точкой локального максимума кривизны границы связана отдельная ветвь скелета. Две области, имеющие несущественные для глаз различия границы, например, за счет шумов, имеют принципиально различные в смысле топологической структуры скелеты.

Поэтому после построения скелета фигуры проводится выделение в скелете части, которая называется базовым скелетом, и удаление элементов, появление которых обусловлено шумовыми эффектами. Метод получения этой информативной части скелета называется регуляризацией скелета с контролируемой точностью.

Объединение скелета фигуры и максимальных вписанных в фигуру окружностей, центры которых и составляют скелет, называется в рамках непрерывного подхода к скелетизации циркулярным представлением фигуры. Объект может быть описан в виде совокупности примитивов определенного вида. В качестве таких примитивов используются жирные линии – объединения однопараметрического семейства кругов переменного радиуса на непрерывных кривых. Жирная линия представляет собой след от перемещения окружности переменного радиуса вдоль непрерывной кривой. Для произвольного растрового бинарного образа вначале строится скелет, затем каждую отдельную ветвь скелета аппроксимируют жирными линиями. Циркулярное представление является развитием понятия скелета и дает возможность не только анализировать форму объекта, но и осуществлять ее преобразования.

На рис. 3 показан многоугольник минимального периметра, скелет (слева), базовый скелет и итоговое циркулярное представление фигуры (справа).

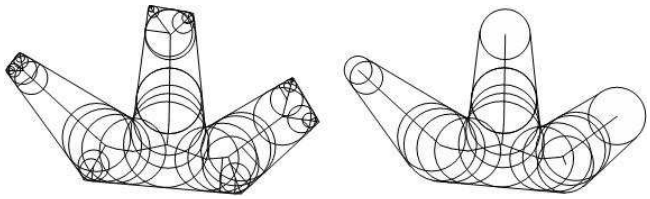


Рис. 3. Многоугольник минимального периметра, скелет (слева), базовый скелет и итоговое циркулярное представление фигуры (справа)

Достоинствами непрерывного подхода к скелетизации являются математическая корректность, адекватность человеческому представлению о форме и ее преобразованиях, широкие возможности для преобразования и сравнения форм, высокая вычислительная эффективность.

Недостатком же является существенное усложнение алгоритмов с точки зрения их математического содержания и программной реализации по сравнению с алгоритмами дискретной скелетизации.

Анализ существующих подходов к скелетизации позволяет сделать предположение о том, что непрерывный подход является наиболее приоритетным для построения математической модели решения задачи сравнения формы бинарных растровых изображений в силу перечисленных выше достоинств.

2. Анализ существующих методов сравнения и классификации формы бинарных растровых изображений

В зависимости от подхода к построению скелета, при разработке методов сравнения фигур скелет рассматривается либо как взвешенный граф, состоящий из вершин и дуг (дискретный подход к построению скелета), либо как циркуляр (непрерывный подход к построению скелета).

Методы, рассматривающие скелет как взвешенный граф, можно условно разделить на использующие для сравнения скелетов топологические признаки графов и использующие редакционное расстояние между двумя графами.

Топологические методы применяются, например, для решения задачи распознавания печатных символов [1]. Каждый контур скелета описывается в виде набора особых точек и так называемого цепного кода, состоящего из точки привязки, числа кодов и массива направлений из очередной точки на следующую точку. Особые точки – это концевые точки

и точки ветвления (триоды), т.е. точки, соседи которых образуют не менее трех связных областей. Для каждой особой точки вычисляются следующие топологические признаки:

— нормированные координаты особой точки (вершины графа);
— длина ребра до следующей вершины в процентах от длины всего графа;
— нормированное направление из данной точки на следующую особую точку;
— нормированное направление входа в точку, выхода из точки (для триодов эти признаки различаются, для точек индекса «1» совпадают с точностью до знака);
— кривизна дуги, точнее «левая» и «правая» кривизна дуги, соединяющей особую точку со следующей вершиной (кривизна слева и справа). Кривизна вычисляется как отношение максимального расстояния от точек дуги (находящихся соответственно слева/справа от прямой) до прямой, соединяющей вершины, к длине отрезка, соединяющей те же вершины.

На рис. 4 условно показаны некоторые из топологических признаков.

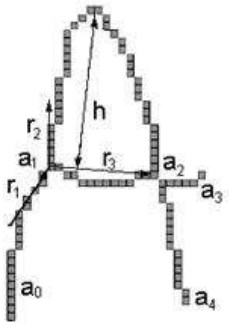


Рис. 4. Примеры скелетных признаков

Обучение метода состоит в построении деревьев распознавания для каждого из определенных заранее (вручную или автоматически) топологических кодов. Распознавание символов ведется на основе обхода построенных деревьев. Метрики для попарного сравнения символов данный метод не предоставляет. Поэтому его нельзя использовать для построения потенциальной функции классификатора по методу опорных векторов (см. следующий раздел).

В методах сравнения формы объектов, использующих редакционное расстояние, в качестве метрики рассматривается соответственно редакционное расстояние (edit distance) между двумя скелетными графами [15].

Понятие редакционного расстояния изначально использовалось для сравнения строк и определяется как минимальное количество операций вставки одного символа, удаления одного символа и замены одного символа на другой, необходимых для превращения одной строки в другую. Чтобы измерить различие между строками, вводится понятие неотрицательных весов для каждой из операций. Следовательно, необходимо найти такую последовательность замен в строке, которая минимизировала бы суммарный вес всех операций [3].

Граф можно представить как множество вершин, дополненное отношением смежности между вершинами, представленное ребрами, соединяющими вершины. Редакционное расстояние для графов определяется аналогично строкам как минимальное количество операций вставки одной вершины, удаления одной вершины и замены одной вершины на другую, вставки одного ребра, удаления одного ребра и замены одного ребра на другое,

необходимых для превращения одного графа в другой. В случае взвешенного редакционного расстояния между графами необходимо найти такую последовательность замен в графе, которая минимизировала бы суммарный вес всех операций. В качестве весов вершин для скелетных графов может употребляться их численно выраженная позиция на изображении.

Задача поиска оптимальной последовательности замен решается методом динамического программирования, но имеет экспоненциальную вычислительную сложность в зависимости от количества вершин в графе. На практике с приемлемой скоростью можно вычислить редакционное расстояние между графами, которые имеют не более 10 вершин [15]. Соответственно, такая метрика не подходит для сравнения изображений сложной формы, в скелетах которых количество вершин превышает 10. Чтобы обойти данное ограничение, для сравнения скелетных графов используются алгоритмы приближенных расчетов редакционного расстояния [14].

В работах, посвященных классификации формы объектов на основе непрерывного скелета – циркулярной модели фигуры, используются меры сходства объектов на основе сравнения составляющих их жирных линий и на основе вычисления расхождения граничных функций ширины двух силуэтов.

Мера сходства объектов на основе сравнения составляющих их жирных линий

Пусть имеется произвольная составная жирная B-сплайновая кривая C, заданная в виде вектор-функции $C(t) = [x(t), y(t), r(t)]$, где $P(t) = [x(t), y(t)]$ задает ось жирной линии, а $r(t)$ – ее ширину, $t \in [0, T]$. Обозначим через $l = f(t)$ функцию длины осевой линии, зависящую от параметра t, тогда $t = f^{-1}(l)$. Рассмотрим функцию $r(l) = r(f^{-1}(l))$ – зависимости ширины жирной кривой от длины осевой линии. Пример жирной линии и соответствующий ей график функции $r(l)$ представлен на рис. 5, где по оси абсцисс отложена длина осевой линии, а по оси ординат – ширина жирной кривой.

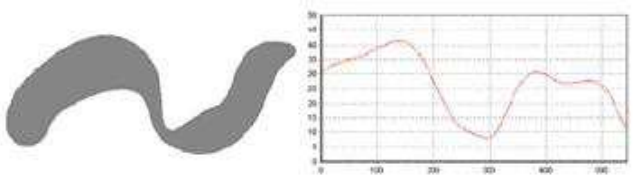


Рис. 5. Жирная B-сплайновая кривая и график зависимости ширины жирной кривой от длины осевой линии

Рассмотрим теперь две произвольные жирные линии $C_1(t)$ и $C_2(t)$. Построим в общей системе координат графики функций $r_1(l)$ и $r_2(l)$. Обозначим через L_1 длину осевой линии жирной кривой $C_1(t)$, а через L_2 – длину осевой линии жирной кривой $C_2(t)$. Пусть $L = \max(L_1, L_2)$, и доопределим функции $r_i(l)$ на $[0, L]$ как $r_i(l) = 0$ при $l > L_i$, если $L_i < L$. Пусть D_1 – множество точек плоскости, ограниченной графиком функции $r_1(l)$ и осью абсцисс, а D_2 - ограниченное графиком $r_2(l)$ и осью абсцисс. Обозначим через $s(D_1 \bigcup D_2)$ – площадь объединения множеств D_1 и D_2, а через $s(D_1 \bigcap D_2)$ – площадь их пересечения. Под мерой сходства двух жирных линий $C_1(t)$ и $C_2(t)$ будем понимать величину

$$\Delta = \frac{s(D_1 \bigcap D_2)}{s(D_1 \bigcup D_2)} = \frac{\int_0^L \min(r_1(l), r_2(l))dl}{\int_0^L \max(r_1(l), r_2(l))dl}$$

.

Поскольку площадь пересечения двух множеств не превышает площади их объединения, имеем $s(D_1 \bigcup D_2) \geqslant s(D_1 \bigcap D_2)$ и величина $\Delta \leqslant 1$. Очевидно, что чем больше значение Δ, тем две жирные линии более похожи. Следует также заметить, что у абсолютно идентичных жирных линий величина сходства будет равна единице. Тем самым, нахождение меры близости двух жирных линий сводится к нахождению отношения двух определенных интегралов [6].

Далее для сравнения образов в конкретных приложениях проводится обработка составляющих их жирных линий. Например, для задачи попарного сравнения образов человеческих ладоней предлагается следующий алгоритм:

1. Для каждого образа строится его циркулярное разложение,
2. Из каждого разложения выбираются и идентифицируются пять самых длинных жирных линий,
3. Производится попарное сравнение соответствующих жирных линий. Результат сравнения i-ой пары есть значение $\Delta_i \in [0,1]$, $i = 1,\ ...,\ 5$, которое характеризует меру сходства двух жирных линий.
4. Результатом сравнения двух образов ладоней будет значение $\omega = \frac{\sum_{i=1}^{5} \Delta_i}{5}$. Полученное значение $\omega \in [0,1]$ показывает величину относительной меры сходства двух образов ладоней [13].

Мера сходства объектов на основе вычисления расхождения граничных функций ширины двух силуэтов

Возьмем точку на скелете и начнем обход скелета по часовой стрелке. Зависимость ширины скелета от длины пройденного пути будем называть *граничной функций ширины*. Обход закончится, когда вернемся в точку, откуда начинали движение. Отмасштабируем фигуру таким образом, чтобы длина полного пути обхода равнялась 1, т.е. будем считать, что граничная функция определена на отрезке [0,1]. В зависимости от того, с какой точки мы начинали движение, получим разные функции, но они будет отличаться друг от друга лишь циклическим сдвигом в области аргумента. Кроме ширины построим аналогичную функцию для степени вершины скелета. Для узловых вершин скелета степень вершины равна количеству исходящих из нее ребер, для всех остальных точек скелета (это точки, которые находятся строго внутри ребра) определим ее равным 2.

Пусть имеется два силуэта, $r_1 : [0,1] \to R^2$ и $r_2 : [0,1] \to R^2$ – граничные функции ширины для них, а $\deg_1 : [0,1] \to \{1,2,3\}$ и $\deg_2 : [0,1] \to \{1,2,3\}$ граничные функции для степени вершины скелетов соответственно. Объединим две такие функции в двухкомпонентный вектор $R_i = (r_i, \deg_i) : i \in \{1,2\}$. Обозначим через T_δ оператор циклического сдвига аргументов на δ для функций, определенных на отрезке [0,1], т.е. $\forall w : [0,1] \to [0,1]$:

$$T_\delta \circ f(t) = \begin{cases} f(t+\delta), & \text{при } t+\delta \leqslant 1 \\ f(t+\delta-1), & \text{при } t+\delta > 1 \end{cases}$$

Здесь значком $\circ$ обозначена операция суперпозиции. Задача выравнивания заключается в построении непрерывного монотонного отображения $w : [0,1] \to [0,1]$ и нахождения сдвига $\delta \geqslant 0$, которые сопоставляют граничные функции ширины $R_1(t) \leftrightarrow T_\delta \circ R_2(w(t))$ и при этом минимизируют расхождение, заданное в виде функционала:

$$d(R_1, R_2) = \min_{\delta \in [0,1]} \min_{\substack{w \in C[0,1] \\ w(0)=0 \\ w(1)=1 \\ w-\text{монот.}}} \int_0^1 \rho\left(R_1(t), R_2\left(T_\delta \circ w(t)\right)\right) \sqrt{1 + w'(t)^2} dt$$

Будем требовать, чтобы узловые вершины скелетов по возможности совпадали (топология скелетных графов может отличаться), для этого в функционал добавлен член $k_1 \cdot |\deg_1 - \deg_2|$. Также потребуем, чтобы граничные функции ширины как можно лучше совпадали после подгонки. В итоге получаем следующую функцию штрафа:

$$\rho = (|r_1 - r_2| + k_1 \cdot |\deg_1 - \deg_2| + k_2)$$

Здесь k_2 константа, которая отвечает за гладкость кривой $w(t)$.

Отметим некоторые свойства выравнивания $R_1(t) \leftrightarrow T_\delta \circ R_2(w(t))$:

1. Очевидно, что выполняется условие рефлексивности: $d(R, R) = 0$.
2. Легко проверить, что такое выравнивание симметрично относительно R_1, R_2, т.е. $d(R_1, R_2) = d(R_2, R_1)$.
3. Однако неравенство треугольника не выполняется: $d(R_1, R_2) + d(R_2, R_3)$ не всегда больше $d(R_1, R_3)$.

Следовательно, $d(R_1, R_2)$ не является метрикой. Тем не менее, значение функционала $d(R_1, R_2)$ показывает, насколько похожи силуэты.

Дискретизируем задачу относительно параметра δ. Выберем N точек $\frac{i}{N}, i = 0..N$ на отрезке [0,1] и будем перебирать значение δ среди них. Тогда задача сводится к N подзадачам, каждая из которых решается методом динамического программирования за время $O(N^2)$. Следовательно, задача минимизации решается за время $O(N^3)$ [11].

3. Задача сравнения форм объектов при помощи потенциальной функции скелетного графа

Наиболее эффективным на сегодняшний день методом классификации данных является метод опорных векторов (Support Vector Machines – SVM). В основу его положен алгоритм построения оптимальной разделяющей гиперплоскости, предложенный в 1963 году В.Н. Вапником [2]. Метод опорных векторов является линейным классификатором, а следовательно, применение его на линейно неразделимых выборках ведет к появлению большого количества неверно классифицированных объектов. Для решения этой проблемы в 1992 году Бернхард Босер, Изабель Гийон и В.Н. Вапник предложили способ создания нелинейного классификатора, в основе которого лежит переход от используемых в SVM скалярных произведений к произвольным ядрам линейного преобразования, так называемый kernel trick (предложенный впервые М.А. Айзерманом, Э.М. Броверманом и Л.В. Розоноером для метода потенциальных функций).

Это переход от исходного пространства признаковых описаний объектов X к новому, так называемому спрямляющему, пространству H с помощью некоторого преобразования $\psi : X \to H$. Если пространство H имеет достаточно высокую размерность, то можно предположить, что в нём выборка окажется линейно разделимой.

Пример перехода к расширенному пространству показан на рис. 6. Точки двух классов (внешняя и внутренняя окружность) на плоскости линейно не разделимы. Если же мы перенесем эти точки на сферу (трехмерное пространство), то тогда они разделяются плоскостью, которая срезает часть сферы вместе с точками на внутренней окружности. Таким образом, «выгнув» пространство вместе с точками с помощью отображения $\psi : X \to H$, можно легко найти разделяющую гиперплоскость.

Результирующий алгоритм похож на алгоритм линейной классификации, с той лишь разницей, что каждое скалярное произведение в методе опорных векторов заменяется

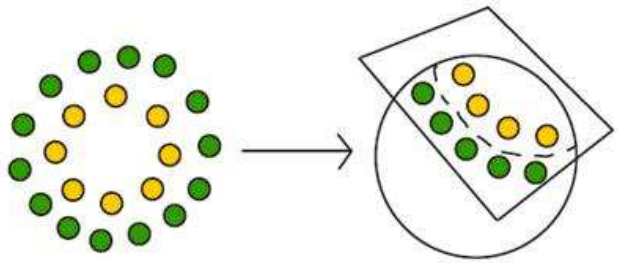


Рис. 6. Пример перехода к расширенному пространству для построения разделяющей гиперплоскости

нелинейной потенциальной функцией (скалярным произведением в пространстве с большей размерностью). В этом пространстве уже может существовать оптимальная разделяющая гиперплоскость. Так как размерность получаемого пространства может быть больше размерности исходного, то преобразование, сопоставляющее скалярные произведения, будет нелинейным, а значит функция, соответствующая в исходном пространстве оптимальной разделяющей гиперплоскости, будет также нелинейной.

Отсюда вытекает естественное требование: пространство H должно быть наделено скалярным произведением, в частности, подойдёт любое евклидово, а в общем случае и гильбертово, пространство.

Определение потенциальной функции (ядра): функция $K: X \times X \to R$ называется *ядром (kernel function)*, если она представима в виде $K: (x, x') = (\psi(x), \psi(x'))$ при некотором отображении $\psi: X \to H$, где H – пространство со скалярным произведением.

Алгоритм классификации по методу опорных векторов зависит только от скалярных произведений объектов, но не от самих признаковых описаний. Это означает, что скалярное произведение (x, x') можно формально заменить потенциальной функцией $K(x, x')$. Поскольку потенциальная функция в общем случае нелинейна, такая замена приводит к существенному расширению множества реализуемых алгоритмов. Более того, можно вообще не строить спрямляющее пространство H в явном виде, и вместо подбора отображения ψ заниматься непосредственно подбором потенциальной функции.

Можно и вовсе отказаться от признаковых описаний объектов. Для такого подхода был придуман термин беспризнаковое распознавание (featureless recognition). Объекты можно изначально задать информацией об их попарном взаимоотношении, например, отношении сходства. Если эта информация допускает представление в виде двуместной функции $K(x, x')$, удовлетворяющей аксиомам скалярного произведения, то задача может решаться методом SVM. Поэтому необходимо найти отношение сходства между двумя объектами, являющееся метрикой.

Функция $K(x, x')$ является ядром тогда и только тогда, когда она симметрична, $K(x, x') = K(x', x)$, и неотрицательно определена:

$\int_X \int_X K(x, x')g(x)g(x')dxdx' \geqslant 0$ для любой функции $g: X \to R$.

Существует эквивалентное определение неотрицательной определённости: функция $K(x, x')$ неотрицательно определена, если для любой конечной выборки $X_p = (x_1, \ldots, x_p)$ из X матрица $K = \|K(x_i, x_j)\|$ размера $p \times p$ неотрицательно определена: $z^T K z > 0$ для любого $z \in R^p$.

Проверка неотрицательной определённости функции в практических ситуациях может оказаться делом нетривиальным. Часто ограничиваются перебором конечного числа функций, про которые известно, что они являются ядрами. Среди них выбирается луч-

шая, как правило, по критерию скользящего контроля. Очевидно, что это не оптимальное решение. На сегодняшний день проблема выбора ядра, оптимального для данной конкретной задачи, остаётся открытой.

Существуют правила порождения, позволяющие строить ядра в практических задачах:

1. Произвольное скалярное произведение $K(x, x') = \langle x, x' \rangle$ является ядром.
2. Константа $K(x, x') = 1$ является ядром.
3. Произведение ядер
   $K(x, x') = K_1(x, x')K_2(x, x')$ является ядром.
4. Для любой функции $\psi : X \to R$ произведение $K(x, x') = \psi(x), \psi(x')$ является ядром.
5. Линейная комбинация ядер с неотрицательными коэффициентами $K(x, x') = \alpha_1 K_1(x, x') + \alpha_2 K_2(x, x')$ является ядром.
6. Композиция произвольной функции $\varphi : X \to X$ и произвольного ядра K_0 является ядром: $K(x, x') = K_0(\varphi(x), \varphi(x'))$.
7. Если $s : X \times X \to R$ - произвольная симметричная интегрируемая функция, то $K(x, x') = \int_X s(x, z)s(x', z)dz$ является ядром.
8. Функция вида $K(x, x') = k(x - x')$ является ядром тогда и только тогда, когда Фурье-образ $F[k](\omega) = (2\pi)^{\frac{n}{2}} \int_X e^{-i\langle \omega, x \rangle} k(x)dx$ неотрицателен.
9. Предел локально-равномерно сходящейся последовательности ядер является ядром.
10. Композиция произвольного ядра K_0 и произвольной функции $f : R \to R$, представимой в виде сходящегося степенного ряда с неотрицательными коэффициентами $K(x, x') = f(K_0(x, x'))$, является ядром. В частности, функции $f(z) = e^z$ и $f(z) = \frac{1}{1-z}$ от ядра являются ядрами.

Наиболее распространённые ядра:

— полиномиальное (однородное)

$$K(x, x') = (x \cdot x')^d$$

— полиномиальное (неоднородное)

$$K(x, x') = (x \cdot x' + 1)^d$$

— радиальная базисная функция

$$K(x, x') = \exp(-\gamma \left\| x - x' \right\|^2),$$

для $\gamma > 0$

— радиальная базисная функция Гаусса

$$K(x, x') = \exp(-\frac{\left\| x - x' \right\|^2}{2\sigma^2})$$

— сигмоид

$$K(x, x') = \tanh(kx \cdot x' + c),$$

для почти всех $k > 0$ и $c < 0$ [10, 16].

На основании материала предыдущего раздела можно утверждать, что для сравнения форм бинарных растровых изображений задача поиска метрик, на основе которых может быть построена потенциальная функция, остается очень актуальной: на данный момент

метрика может быть получена только методом редакционного расстояния, но этот метод учитывает преимущественно количество вершин и ребер в скелетном графе, опуская другие его не менее важные особенности, как например, ширина или кривизна ветвей. Следовательно, часть признаков объектов не будет использована для построения классификатора, что ведет к понижению надежности распознавания.

Рассмотрим в качестве примера одну из потенциальных функций, которая может быть построена на основе редакционного расстояния между графами.

Пусть X – признаковое пространство скелетных графов и $X^t \subset X$ – обучающая выборка образов. Для фиксированного образа $x_0 \in X^t$ потенциальная функция $k_{x_0} : X \times X \to R$ может быть определена как

$$k(x, x') = k_{x_0}(x, x') = \frac{1}{2}(d(x, x_0)^2 + d(x_0, x')^2 - d(x, x')^2),$$

где $d(\cdot, \cdot)$ – редакционное расстояние между двумя образами.

Эта потенциальная функция может быть интерпретирована как мера квадратного расстояния от образа x до x_0 и от образа x_0 до x' относительно к квадратному расстоянию от образа x до x' напрямую.

Данная потенциальная функция действительна, потому что существует пространство со скалярным произведением H, где каждый граф $x \in X$ представлен единственным вектором $\Phi(x) \in H$ и скалярное произведение векторов эквивалентно потенциальной функции. Образ x_0 называется нулевым графом, поскольку он обладает свойствами начала координат признакового пространства скелетных графов.

Эксперименты, проведенные авторами данной потенциальной функции, показывают, что применение даже такой простой функции в методе опорных векторов дает более корректные результаты классификации изображений по сравнению с методом ближайших соседей [15]. Так что поиск метрик в пространстве скелетных графов и построение потенциальных функций на их основе представляется очень перспективным направлением для классификации изображений на основе скелетного представления их формы.

Выводы

Для сравнения и классификации форм бинарных растровых изображений задача поиска метрик в пространстве скелетных графов остается очень актуальной: на данный момент метрика может быть получена только методом редакционного расстояния, но этот метод учитывает преимущественно количество вершин и ребер в скелетном графе, опуская другие его не менее важные особенности, как например, ширина или кривизна ветвей. Следовательно, часть признаков объектов не будет использована для построения классификатора, что ведет к понижению надежности распознавания.

Наиболее перспективной представляется метрика для непрерывных скелетов фигуры, которые описываются скелетным графом и вписанными в фигуру максимальными окружностями с центрами на скелете, поскольку такое математическое описание более адекватно человеческому представлению о форме и ее преобразованиях. Построенная на основе такой метрики функция ядра должна повысить возможности использующего ее классификатора.

Поскольку такой метрики в настоящее время нет, то актуальна задача ее нахождения и построения на ее основе потенциальной функции, применяемой в методе опорных векторов для классификации объектов.

Литература

[1] *Афонасенко А. В., Елизаров А. И.* Обзор методов распознавания структурированных символов. // Доклады ТУСУРа, 2008, № 2, часть 1.

[2] *Вапник В. Н., Червонекис А. Я.* Об одном классе алгоритмов обучения распознаванию образов. // Автоматика и телемеханика, 1964, № 4.

[3] *Гасфилд Д.* Строки, деревья и последовательности в алгоритмах: Информатика и вычислительная биология. // СПб: Невский диалект; БХВ-Петербург, 2003.

[4] *Гонсалес Р., Вудс Р.* Цифровая обработка изображений. // М: Техносфера, 2005.

[5] *Местецкий Л. М.* Непрерывная морфология бинарных изображений: фигуры, скелеты, циркуляры. // М: ФИЗМАЛИТ, 2009.

[6] *Местецкий Л. М.* Компьютерная графика на основе жирных линий. // Труды межд. конф. «Графикон-2000», Москва, 2000.

[7] *Местецкий Л. М., Рейер И. А.* Непрерывная гранично-скелетная модель дискретного изображения с контролируемой точностью аппроксимации. // Доклады Всероссийской конференции «Математические методы распознавания образов», (ММРО-11). Москва, 2003, с. 367-371.

[8] *Моттль В. В.* Метрические пространства, допускающие введение линейных операций и скалярного произведения. // Доклады Академии наук, 2003, том 388, № 3, с.1-4.

[9] Применение волнового алгоритма для нахождения скелета растрового изображения. // [Электронный ресурс]. Режим доступа: http://ocrai.narod.ru. Загл. с экрана.

[10] *Середин О. С., Моттль В. В.* Методы беспризнакового распознавания образов // Тула: Изд-во ТулГУ, 2004.

[11] *Цискаридзе А. К.* Восстановление пространственных циркулярных моделей по силуэтным изображениям // Диссертация на соискание ученой степени кандидата физико-математических наук. МФТИ, 2010.

[12] *Dupe F. X., Brun L.* Shape classification using a flexible graph kernel // Proceedings of Academic Press 2009 , September 2009.

[13] *Mestetskiy L., Semenov A.* Palm Shape Comparison Based on Fat Curves // Proceedings of 7th International conference on Pattern recognition and image analysis: new information technologies, St. Petersburg, 2004, pp. 788–791.

[14] *Neuhaus M., Bunke H.* An error-tolerant approximate matching algorithm for attributed planar graphs and its application to fingerprint classification, in: Proceedings of the 10th International Workshop on Structural and Syntactic Pattern Recognition. // Lecture Notes in Computer Science, vol. 3138, Springer, Berlin, 2004, pp. 180–189.

[15] *Neuhaus M., Bunke H.* Edit-distance based kernel for structural pattern classification. // Pattern Recognition 39 (2006), pp. 1852–1863.

[16] *Vapnik V.* Statistical Learning Theory. // NY.: J. Wiley, 1998. 768 p.

Semiparametric estimation of the signal subspace*

***Denis Belomestny*[1], *Vladimir Panov*[2], *Vladimir Spokoiny*[3]**

1 — Laboratory for Structural Methods of Data Analysis in Predictive Modeling (MIPT) and University Duisburg-Essen, denis.belomestny@uni-due.de

2 — Laboratory for Structural Methods of Data Analysis in Predictive Modeling (MIPT) and University Duisburg-Essen, vladimir.panov@uni-due.de

3 — Laboratory for Structural Methods of Data Analysis in Predictive Modeling (MIPT) and Weierstrass Institute for Applied Analysis and Stochastics, spokoiny@wias-berlin.de

Let a high-dimensional random vector $\vec{X}$ be represented as a sum of two components — a signal $\vec{S}$ that belongs to some low-dimensional linear subspace $\mathcal{S}$, and a noise component $\vec{N}$. This paper presents a new approach for estimating the subspace $\mathcal{S}$ based on the ideas of the Non-Gaussian Component Analysis. Our approach avoids the technical difficulties that usually appear in similar methods — it requires neither the estimation of the inverse covariance matrix of $\vec{X}$ nor the estimation of the covariance matrix of $\vec{N}$.

Keywords: *dimension reduction, non-Gaussian components, signal subspace*

Introduction

Assume that a high-dimensional random variable $\vec{X} \in \mathbb{R}^d$ be represented as a sum of two independent components — a low-dimensional signal (which one can imagine as "a useful part" or "an information") and a noise component with a Normal distribution. More precisely,

$$\vec{X} = \vec{S} + \vec{N}, \tag{1}$$

where $\vec{S}$ belongs to some low-dimensional linear subspace $\mathcal{S}$, $\vec{N}$ is a normal vector with zero mean and unknown covariance matrix, and $\vec{S}$ is independent of $\vec{N}$. This structural assumption follows the observation that in applications the "useful part" is non-Gaussian while the "rest part" can be interpreted as a high-dimensional noise. For the sake of simplicity, we assume that the expectation of $\vec{X}$ vanishes and the covariance matrix of $\vec{X}$, which is denoted by Σ, is non-degenerated.

Denote the dimension of $\vec{S}$ by m. In this paper, m is fixed such that the representation (1) is unique; the existence of such m is proved under some mild assumptions by Theis and Kawanabe (2007). If $m = 0$, then our model reduces to the pure-parametric case; if $m = d$, then the model is pure-nonparametric. Obviously, the representation (1) links pure Gaussian (PCA) and pure non-Gaussian (ICA) modelling.

The aim of this paper is to estimate vectors from the subspace $\mathcal{S}$, which we call *the signal subspace*. A very related task, estimation of so called *the non-Gaussian subspace* $\mathcal{I}$ (the definition will be given below) has been extensively studied in the literature. The original method known as Non-Gaussian Component Analysis (NGCA) was proposed by Blanchard et al. (2006), and later improved by Dalalyan et al. (2007), Kawanabe et al. (2007), Diederichs et al. (2010).

In almost all papers mentioned above, the problem of estimation of the vectors from $\mathcal{S}$ is not considered in details; natural estimators require the estimation of the covariance matrix of the noise. Moreover, practical usage meets another technical problem — the estimation of the

The authors are partially supported by Laboratory for Structural Methods of Data Analysis in Predictive Modeling, MIPT, RF government grant, ag. 11.G34.31.0073.

inverse covariance matrix of $\vec{X}$. Each of these tasks is an obstacle in real-world applications of the method. In this article, we propose a new approach, which avoids the mentioned problems.

The paper is organized as follows. In the second section, we formulate and discuss the mathematical statements that yield a method for estimating the vectors from $\mathcal{S}$. We start with Theorem 1, which implies a special representation for the density function of $\vec{X}$. This representation is new and plays a central role in the validation of the method. The theoretical base of our approach is given in Theorem 2 and Lemma 3. Section 3 contains the full description of the algortihm. Next, we discuss different representations for the density function and point out the advantages of our formulation. The last section states what has been done before and what is the contribution of the present paper. All proofs are collected in Appendix A; some additional information about the NGCA methodology is given in Appendix B.

Theoretical base for the estimation of the signal subspace

This section presents the theoretical results that are needed for our purposes. The proofs are collected in Appendix A.

The first theorem gives the semiparametric density representation for the random vector $\vec{X}$. Such facts are known in the literature (see e.g. Blanchard et al., 2006) but the formulation given below principally differs from the previous versions, see the next section for discussion.

Theorem 1. Let the structural assumption (1) be fulfilled. Then the density function of the random vector $\vec{X}$ can be represented as follows:

$$p(\vec{x}) = \mathbf{g}(\mathbf{T}\vec{x})\; p^{\mathcal{N}}(\vec{x}, \Sigma), \tag{2}$$

where

— $\mathbf{T}:\ \mathbb{R}^d \to \Sigma^{-1/2}\mathcal{S}$ is the linear transformation defined as the projection of the vector $\Sigma^{-1/2}\vec{x}$ on the subspace $\Sigma^{-1/2}\mathcal{S}$, i.e.,

$$\mathbf{T}\vec{x} := \Pr{}_{\Sigma^{-1/2}\mathcal{S}}\{\Sigma^{-1/2}\vec{x}\}, \tag{3}$$

— $\mathbf{g}:\ \Sigma^{-1/2}\mathcal{S} \to \mathbb{R}$ is defined by

$$\mathbf{g}(\vec{t}\,) = \frac{q\,(\vec{t}\,)}{p^{\mathcal{N}}(\vec{t}, \mathbf{I}_m)}, \tag{4}$$

and $q(\cdot)$ is the density function of the random variable $\mathbf{T}\vec{X}$.

As it was mentioned in the introduction, our aim is to recover the subspace $\mathcal{S}$. This can be done thanks to the following theorem, which is a new result in the context of the Non-Gaussian Component Analysis.

Theorem 2. Let $\mathbf{T}$ be the linear transformation defined by (3). Then

$$\mathcal{S} = \Sigma\ (\operatorname{Ker}\mathbf{T})^{\perp}. \tag{5}$$

In Blanchard et al. (2006), a transformation $\mathcal{T}$ is considered instead of $\mathbf{T}$:

$$\mathcal{T}\vec{x} := \Pr{}_{\Gamma^{-1/2}\mathcal{S}}\{\Gamma^{-1/2}\vec{x}\}, \tag{6}$$

where by Γ we denote the covariance matrix of the noise component. In the paper by Blanchard et al., the subspace $(\operatorname{Ker}\mathcal{T})^{\perp}$ is called *the non-Gaussian subspace* and is in fact the main object of interest. We would like to stress here that $\mathcal{T} \neq \mathbf{T}$, and equalities like (5) are wrong for $\mathcal{T}$.

One of the main results about the NGCA approach gives the practical method for estimating vectors from $(\operatorname{Ker}\mathcal{T})^{\perp}$. Similar result can be formulated also for the subspace $(\operatorname{Ker}\mathbf{T})^{\perp}$.

Lemma 3. Assume that the structural assumption (1) is fulfilled. Then for any function $\psi \in \mathcal{C}^1(\mathbb{R}^d, \mathbb{R})$ there exists a vector $\beta \in (\text{Ker } \mathbf{T})^\perp$ such that

$$\mathbb{E}\left(\nabla\psi(\vec{X})\right) - \beta = \Sigma^{-1}\mathbb{E}\left(\vec{X}\psi(\vec{X})\right). \tag{7}$$

Corollary 4. Let the structural assumption (1) be fulfilled and let a function $\psi \in \mathcal{C}^1(\mathbb{R}^d, \mathbb{R})$ be such that $\mathbb{E}\left(\vec{X}\psi(\vec{X})\right) = 0$. Then

$$\mathbb{E}\left(\nabla\psi(\vec{X})\right) \in (\text{Ker } \mathbf{T})^\perp.$$

Theorem 2 and Lemma 3 yield a method for estimating vectors from the subspace $\mathcal{S}$ that is described in the next section.

Algorithm for the estimation of the signal subspace

The first step is to estimate vectors from the subspace $(\text{Ker } \mathbf{T})^\perp$ using Lemma 3. Theoretically, the best way for the estimation is to find a set of functions $\{\psi\}$ such that $\mathbb{E}\left(\vec{X}\psi(\vec{X})\right) = 0$ for any element of this set, and to estimate the vectors from $(\text{Ker } \mathbf{T})^\perp$ by $\mathbb{E}\left(\nabla\psi(\vec{X})\right)$, see Corollary 4. In practice, such set $\{\psi\}$ is unknown; usually it is more realistic to consider some ψ such that $\mathbb{E}\left(\vec{X}\psi(\vec{X})\right)$ is close to zero (but not exactly zero). In this case, according to Lemma 3, the vector $\mathbb{E}\left(\nabla\psi(\vec{X})\right)$ is close to some vector from the subspace $(\text{Ker } \mathbf{T})^\perp$.

We suggest to use the convex projection method, see Diederichs (2007) and Diederichs et al. (2010). Since the description of this method requires at least one page, and its discussion is not an objective of the article (and merits a separate publication), we put the explanation in the appendix of the paper.

The second step. Denote the vectors obtained on the first step by $\vec{\beta}_k$, $k = 1..K$. Now one can use Theorem 2 and estimate vectors from the signal subspace by $\vec{\gamma}_k := \hat{\Sigma}\vec{\beta}_k$, where $\hat{\Sigma}$ is an estimator of the matrix Σ,

$$\hat{\Sigma} := n^{-1}\sum_{i=1}^{n} \vec{X}_i\vec{X}_i^\top.$$

The third step. The next problem is how to utilize the set $\{\vec{\gamma}_k\}_{k=1}^K$ for recovering the target space. According to the method SAMM (structural adaptation via maximum minimization method, see Dalalyan et al. (2007)), the "optimal" projector on the target subspace is defined as follows:

$$\widehat{\Pi}_{NG} := \underset{\left\{\substack{\Pi:\Pi^*=\Pi \\ 0\leqslant\Pi\leqslant I \\ \mathrm{tr}(\Pi)\leqslant m}\right\}}{\arg\min} \max_{1\leqslant k\leqslant K} \vec{\gamma_k}^\top(I-\Pi)\vec{\gamma}_k. \tag{8}$$

Remark 1. Note that the inverse covariance matrix is included in the formula (7) but our approach doesn't require the estimation of it. In fact, Lemma 3 is used only for theoretical justification of the first step; practical method described above needs neither the estimation of Σ^{-1} nor the estimation of Γ. On the second step, one uses only the representation (5), which also allows to avoid the estimation of the inverse covariance matrix. The method SAMM doesn't require the estimation of the inverse covariance matrix and therefore can be used in this algorithm.

Representations for the density function of $\vec{X}$

The proofs of Theorem 2 and Lemma 3 are based on the special representation of the density function of $\vec{X}$ that is given in Theorem 1. Similar representations have been studied extensively in previous papers about NGCA. Such facts are usually stated in the following form: if structural assumption (1) is fulfilled, then the density function of a random vector $\vec{X} \in \mathbb{R}^d$ can be represented as

$$p(\vec{x}) = g(T\vec{x})\varphi_A(\vec{x}), \tag{9}$$

where $T : \mathbb{R}^d \to \mathcal{E}$ is a linear transformation ($\mathcal{E}$ — some subspace with $\dim\mathcal{E} = m$), $g : \mathcal{E} \to \mathbb{R}$ — a function, and A — a $d \times d$ symmetric positive matrix. In most papers, formula (9) is proven only for $A = \Gamma$, see e.g. Kawanabe et al. (2007). Another way is to start from the representation (9) without giving the motivation in the spirit of (1), see e.g. Blanchard et al. (2006). In this respect, the result of Theorem 1 can be briefly explained as follows: for any $\vec{X}$ in the form (1), one can find a function g such that (9) is fulfilled with $T = \mathbf{T}$ and $A = \Sigma$.

The existence of a representation in the form (9) can be easily shown in the following way. Note that the model (1) is equivalent to a linear mixing model

$$\vec{X} = A_S\vec{X}_S + A_N\vec{X}_N, \tag{10}$$

where $\vec{X}_S \in \mathbb{R}^m, \vec{X}_N \in \mathbb{R}^{d-m}$ are two random variables; $\vec{X}_N$ is a normal vector with unknown covariance matrix; $\vec{X}_S$ is independent of $\vec{X}_N$; $A_S \in \mathrm{Matr}(d \times m), A_N \in \mathrm{Matr}(d \times (d-m))$ are two deterministic matrices such that columns of these matrices are independent. In this formulation, the signal subspace is spanned by the columns of matrix A_S.

Therefore the vector X is a linear transformation of the vector $\vec{X}' := (\vec{X}_S; \vec{X}_N)$ (this notation means that $\vec{X}'$ is a concatenation of $\vec{X}_S$ and $\vec{X}_N$). This yields that $p(x) \propto g(\vec{X}_S)\varphi(\vec{X}_N)$, where by g we denote the density function of the m-dimensional non-Gaussian component, and by φ — the density function of the normally distributed random variable $\vec{X}_N$. Thus, the representation (9) is proven with some A and T.

As it was mentioned before, the precise formulation for the density of $\vec{X}$ that is needed for our purposes is given in Theorem 1.

Conclusion

The NGCA method has been already widely discussed in the literature (see the introduction). A substantial difference between the previous papers and this article lies in the object of estimation: Blanchard et al, (2006), Diederichs et al. (2010) and other authors concentrate on the estimation of the subspace $\mathcal{T}$ (defined by (6)), and we aim to estimate the subspace $\mathbf{T}$ (see (3)). The estimation of $\mathcal{T}$ also relies on Lemma 3, and this lemma can be found in previous articles. However, the second step is quite new for the NGCA approach. Our methodology (estimation of the signal subspace without estimation of the inverse covariance matrix) has not been previously discussed in the literature. The mathematical base for the second step is given in Theorem 2, which is also new.

Our method relies on the special representation for the density function, which is discussed in Section 4. I would like to stress here that the representation (2) with the operator $\mathbf{T}$ in the form (3) is also a new technical result.

Acknowledgments

The authors are partially supported by Laboratory for Structural Methods of Data Analysis in Predictive Modeling, MIPT, RF government grant, ag. 11.G34.31.0073.

Appendix A. Proofs of the main theoretical facts

Proof of Theorem 1

1. Denote

$$\vec{X}' := \Sigma^{-1/2}\vec{X} = \Sigma^{-1/2}\vec{S} + \Sigma^{-1/2}\vec{N}, \tag{11}$$

and introduce the notation $\vec{S}' = \Sigma^{-1/2}\vec{S}, \quad \vec{N}' = \Sigma^{-1/2}\vec{N}$.

The first component in (11) belongs to the subspace $\mathcal{S}' := \Sigma^{-1/2}\mathcal{S}$. Denote by $\mathcal{N}'$ the subspace that is orthogonal to $\mathcal{S}'$; one can show $\mathcal{N}'$ coincides with the subspace $\Sigma^{1/2}\mathcal{S}^{\perp}$ (Sugiyama et al, 2008).

Vector $\vec{N}'$ can be decomposed into the sum of two vectors, $\vec{N}' = \vec{N}_{\mathcal{S}'} + \vec{N}_{\mathcal{N}'}$, where $\vec{N}_{\mathcal{S}'} \in \mathcal{S}'$, $\vec{N}_{\mathcal{N}'} \in \mathcal{N}'$. This yields that $\vec{X}$ is represented by

$$\vec{X}' = \underbrace{\vec{S}' + \vec{N}_{\mathcal{S}'}}_{\in\,\mathcal{S}'} + \underbrace{\vec{N}_{\mathcal{N}'}}_{\in\,\mathcal{N}'}.$$

Let us choose the basis of $\mathbb{R}^d$ such that the first m vectors $\vec{v}_1, ..., \vec{v}_m$ belong to the subspace $\mathcal{S}'$, while $\vec{v}_{m+1}, ..., \vec{v}_d$ belong to $\mathcal{N}'$. Decompose the vectors $\vec{Z}' := \vec{S}' + \vec{N}_{\mathcal{S}'}$ and $\vec{N}_{\mathcal{N}'}$ into the basis $\vec{v}_1, .., \vec{v}_d$:

$$\vec{Z}' = \sum_{i=1}^{m} z_i \vec{v}_i, \quad \vec{N}_{\mathcal{N}'} = \sum_{i=m+1}^{d} n_i \vec{v}_i, \tag{12}$$

where all coefficients z_i and n_i are random. Since $\vec{X}'$ is a standardized vector,

$$\mathbf{I}_d = \mathbb{E}\left[\vec{X}'\vec{X}'^{\top}\right] = \sum_{i,j=1}^{m} \mathbb{E}\left[z_i z_j\right] \vec{v}_i \vec{v}_j^{\top} + \sum_{i=1}^{m}\sum_{j=m+1}^{d} \mathbb{E}\left[z_i n_j\right] \vec{v}_i \vec{v}_j^{\top} + \sum_{i,j=m+1}^{d} \mathbb{E}\left[n_i n_j\right] \vec{v}_i \vec{v}_j^{\top},$$

and we conclude that the vectors $\vec{Z}'$ and $\vec{N}_{\mathcal{N}'}$ are also standardized.

2. Denote by $F'(\vec{x}')$ and $p'(\vec{x}')$ the distribution function and the density function of the vector $\vec{X}'$,

$$F'(\vec{x}') = \mathbb{P}\left\{\vec{X}' \leqslant \vec{x}'\right\} = \mathbb{P}\left\{\vec{Z}' + \vec{N}_{\mathcal{N}'} \leqslant \vec{x}'\right\}. \tag{13}$$

Note that the vectors $\vec{S}' = \Sigma^{-1/2}\vec{S}$ and $\vec{N}' = \Sigma^{-1/2}\vec{N}$ are independent as the functions of the independent vectors $\vec{S}$ and $\vec{N}$. Therefore, the vectors $\vec{S}'$, $\vec{N}_{\mathcal{N}'}$ and $\vec{N}_{\mathcal{S}'}$ are mutually (not just pairwise) independent. Finally, $\vec{Z}'$ and $\vec{N}_{\mathcal{N}'}$ are independent as functions of independent variables.

Next, the choice of the basis allows to split the inequality

$$\vec{Z}' + \vec{N}_{\mathcal{N}'} \leqslant \vec{x}' = \sum_{i=1}^{d} x_i \vec{v}_i$$

into two

$$\vec{Z}' \leqslant \sum_{i=1}^{m} x_i \vec{v}_i =: \vec{x}_{\mathcal{S}'}, \quad \vec{N}_{\mathcal{N}'} \leqslant \sum_{i=m+1}^{d} x_i \vec{v}_i =: \vec{x}_{\mathcal{N}'},$$

and to represent the function F' as

$$F'(\vec{x}') = \mathbb{P}\left\{\vec{Z}' + \vec{N}_{\mathcal{N}'} \leqslant \vec{x}'\right\} = \mathbb{P}\left\{\vec{Z}' \leqslant \vec{x}_{\mathcal{S}'}, \vec{N}_{\mathcal{N}'} \leqslant \vec{x}_{\mathcal{N}'}\right\} = \mathbb{P}\left\{\vec{Z}' \leqslant \vec{x}_{\mathcal{S}'}\right\} \mathbb{P}\left\{\vec{N}_{\mathcal{N}'} \leqslant \vec{x}_{\mathcal{N}'}\right\}. \tag{14}$$

Taking the derivatives in (14), we arrive at the representation for the density function of $\vec{X}'$.

$$p'(\vec{x}') = q(\vec{x}_{\mathcal{S}'}) \cdot p^{\mathcal{N}}(\vec{x}_{\mathcal{N}'}, \mathbf{I}_{d-m}) = \frac{q(\vec{x}_{\mathcal{S}'})}{p^{\mathcal{N}}(\vec{x}_{\mathcal{S}'}, \mathbf{I}_m)} \cdot p^{\mathcal{N}}(\vec{x}', \mathbf{I}_d) = \frac{q(\Pr_{\mathcal{S}'}\vec{x}')}{p^{\mathcal{N}}(\Pr_{\mathcal{S}'}\vec{x}', \mathbf{I}_m)} \cdot p^{\mathcal{N}}(\vec{x}', \mathbf{I}_d),$$

where by $q(\cdot)$ we denote the density function of the random vector $\vec{Z}' = \vec{S}' + \vec{N}_{\mathcal{S}'} = \Pr_{\mathcal{S}'}\{\vec{X}'\}$.

3. We complete the proof by deriving representation for the density function of $\vec{X} = \Sigma^{1/2}\vec{X}'$ from the density function of $\vec{X}'$. According to the well-known formula for the density transformation,

$$p(\vec{x}) = \det\left(\Sigma^{-1/2}\right)\, p'(\Sigma^{-1/2}\vec{x}) = \det\left(\Sigma^{-1/2}\right) \cdot \frac{q\Big(\Pr_{\mathcal{S}'}\left\{\Sigma^{-1/2}\vec{x}\right\}\Big)}{p^{\mathcal{N}}\Big(\Pr_{\mathcal{S}'}\left\{\Sigma^{-1/2}\vec{x}\right\}, \mathbf{I}_m\Big)} \cdot p^{\mathcal{N}}(\Sigma^{-1/2}\vec{x}, \mathbf{I}_d).$$

The remark $p^{\mathcal{N}}(\Sigma^{-1/2}\vec{x}, \mathbf{I}_d) = \det\left(\Sigma^{1/2}\right) p^{\mathcal{N}}(\vec{x}, \Sigma)$ concludes the proof.

Proof of Lemma 3. We obtain a more general result:

Lemma 5. Assume that the density function of a random vector $\vec{X} \in \mathbb{R}^d$ be represented in the form (9), where $T : \mathbb{R}^d \to \mathcal{E}$ is any linear transformation ($\mathcal{E}$ — any linear space), $g : \mathcal{E} \to \mathbb{R}$ — any function, and A — any $d \times d$ symmetric positive-defined matrix. Then for any function $\psi \in \mathcal{C}^{(1)}(\mathbb{R}^d, \mathbb{R})$, the vector

$$\vec{\beta} := \mathbb{E}\left[\nabla\psi(\vec{X})\right] - A^{-1}\mathbb{E}\left[\vec{X}\psi(\vec{X})\right] \tag{15}$$

belongs to the subspace $(\mathrm{Ker}\ T)^{\perp}$.

Proof. Since for any function ψ and for any $u \in \mathbb{R}^d$,

$$\int \psi(x+u)p(x)dx = \int \psi(x)p(x-u)dx,$$

we conclude that under some mild assumptions,

$$\mathbb{E}\left[\nabla\psi(\vec{X})\right] = \int \nabla\left[\psi(\vec{x})\right] p(\vec{x})dx = -\int \psi(\vec{x})\nabla\left[p(\vec{x})\right] d\vec{x}. \tag{16}$$

The gradient of the density function can be represented by the sum of two components:

$$\begin{aligned} \nabla p(\vec{x}) &= \nabla\left[\log p(\vec{x})\right] p(\vec{x}) \\ &= \nabla\left[\log g(T\vec{x})\right] p(\vec{x}) + \nabla\left[\log p^{\mathcal{N}}(\vec{x}, A)\right] p(\vec{x}). \end{aligned} \tag{17}$$

We have

$$\begin{aligned} \nabla\left[\log g(T\vec{x})\right] p(\vec{x}) &= \frac{\nabla g(T\vec{x})}{g(T\vec{x})} p(\vec{x}) = \nabla\left[g(T\vec{x})\right] p^{\mathcal{N}}(\vec{x}, A) \\ &= p^{\mathcal{N}}(\vec{x}, A) \cdot T^{\top} \nabla_{\vec{y}}\left[g(\vec{y})\right]|_{\vec{y}=T\vec{x}}, \\ \nabla\left[\log p^{\mathcal{N}}(\vec{x}, A)\right] p(\vec{x}) &= -A^{-1}\vec{x}p(\vec{x}). \end{aligned}$$

Denote

$$\vec{\Lambda} = -\int \psi(\vec{x}) \cdot p^{\mathcal{N}}(\vec{x}, A) \cdot \nabla_{\vec{y}}\left[g(\vec{y})\right]|_{\vec{y}=T\vec{x}}\, d\vec{x}.$$

Then (15) follows with $\vec{\beta} := T^\top \vec{\Lambda} \in (\mathrm{Ker}\ T)^\perp$ because (16) together with (17) yield

$$\mathbb{E}\left[\nabla\psi(\vec{X})\right] = T^\top \vec{\Lambda} + A^{-1}\mathbb{E}\left[\vec{X}\psi(\vec{X})\right].$$

Proof of Theorem 2

The proof is straightforward:

$$\begin{aligned}\mathrm{Ker}\,\mathbf{T} &= \left\{\vec{x}:\ \Sigma^{-1/2}\vec{x} \perp \Sigma^{-1/2}\mathcal{S}\right\} \\ &= \left\{\vec{x}:\ \exists \vec{s} \in \mathcal{S} \mid \vec{x}^\top \left(\Sigma^{-1/2}\right)^\top \Sigma^{-1/2}\vec{s} = 0\right\} = \left\{\vec{x}:\ \exists \vec{s} \in \mathcal{S} \mid \vec{x}^\top\ \Sigma^{-1}\vec{s} = 0\right\} \\ &= \left\{\vec{x}:\ \vec{x} \perp \Sigma^{-1}\mathcal{S}\right\}.\end{aligned}$$

Appendix B. Choice of ψ by the convex projection method

The appendix briefly explains the method of estimation of the functions $\{\psi\}$ called the convex projection method (Diederichs (2007) and Diederichs et al. (2010)). The method gives an algorithm for finding one function ψ; one can repeat the algorithm with different parameters and receive the whole set $\{\psi\}$.

The core of the method is the choice of the function ψ in the following form:

$$\psi(\vec{x}) := \sum_{j=1}^{J} c_j \psi_j(\vec{x}); \qquad \psi_j(\vec{x}) = f(\vec{\omega}_j^T \vec{x}) e^{-||\vec{x}||^2/2}, \tag{18}$$

where f can be any smooth function; for the numerical simulations, Diederichs uses $f(z) = = f_1(z)$ or $f(z) = (1+z^2)^{-1}e^z$; "directions"$\{\omega_j, j = 1..J\}$ are preliminary estimated through the Monte-Carlo sampling from the uniform distribution on the sphere; $\vec{c} = \{c_j, j = 1..J\}$ is a vector from the $\mathscr{L}_1$ - unit ball, which is the object of estimation.

Corollary 4 yields that for any function $\psi \in \mathcal{C}^{(1)}(\mathbb{R}^d, \mathbb{R})$ such that $\mathbb{E}\left[\vec{X}\psi(\vec{X})\right] = 0$, the vector $\mathbb{E}\left[\nabla\psi(\vec{X})\right]$ belongs to $(\mathrm{Ker}\ \mathbf{T})^\perp$. Changing the mathematical expectations in this statement by their empirical counterparts, we conclude that if the vector $\vec{c}$ is chosen such that

$$\gamma(\vec{c}) := \frac{1}{JN}\sum_{j=1}^{J} c_j \left(\sum_{i=1}^{N} \psi_j(\vec{X}_i)\vec{X}_i\right) \approx \vec{0}, \tag{19}$$

then

$$\vec{\beta}(\vec{c}) := \frac{1}{JN}\sum_{j=1}^{J} c_j \left(\sum_{i=1}^{N} \nabla\psi_j(\vec{X}_i)\right) \tag{20}$$

can be considered as the estimate of a vector from $(\mathrm{Ker}\ \mathbf{T})^\perp$. Diederichs (2007) proposes to estimate the coefficient vector $\vec{c}$ by solving the optimization problem

$$(\hat{c}_1, \hat{c}_2, ..., \hat{c}_J) := \arg\min_{\vec{c}}\left\{\|\vec{\xi} - \vec{\beta}(\vec{c})\|_2 \;\middle|\; \gamma(\vec{c}) = \vec{0},\ \|\vec{c}\|_1 \leqslant 1\right\}, \tag{21}$$

where $\vec{\xi}$ is a unit vector, which we call *a probe vector*. Afterwards the estimate of the function ψ is set to $\psi(\vec{x}) := \sum_{j=1}^{J} \hat{c}_j \psi_j(\vec{x})$.

See the article by Diederichs et al. (2007) for examples.

Another popular method for finding ψ was introduced by Blanchard et al. (2006). We don't present that method here because it uses the estimator of the inverse covariance matrix Σ^{-1} and contradicts the philosophy of this paper.

Литература

[1] Belomestny D., Spokoiny V. (2007). Spatial aggregation of local likelihood estimates with applications to classification. *Ann. Statist.* **35**, 2287–2311.

[2] Blanchard, G., Kawanabe, M., Sugiyama, M., Spokoiny, V., Müller, K.-R. (2006). In search of non-Gaussian components of a high-dimensional distribution. *J. Mach. Learn. Res.* **6**, 247–282.

[3] Cook, R.D. (1998). Principal hessian directions revised. *J. Am. Statist. Ass.* **93**, 85-100.

[4] Dalalyan, A., Juditsky, A., Spokoiny, V. (2007). A new algorithm for estimating the effective dimension — reduction subspace. *J. Mach. Learn. Res.* **9**, 1647–1678.

[5] Diaconis, P., and Friedman, D. (1984). Asymptotics of graphical projection pursuit. *Ann.Statist.* **12(3)**, 793–815.

[6] Diederichs, E. (2007). *Semi-parametric reduction of dimensionality.* Ph.D. thesis. Free University of Berlin.

[7] Diederichs, E., Juditsky, A., Spokoiny, V., Schütte, C. (2010). Sparse non-Gaussian component analysis. *IEEE Trans. Inf. Theory.* **15**, 3033–3047.

[8] Hall, P. (1989). Projection pursuit methods. *Ann. Statist.* **17**, 589–605.

[9] Hristache, M., Juditsky A., Polzehl, J., Spokoiny, V. (2001). Structure adaptive approach for dimension reduction. *Ann. Statist.* **6**, 1537–1566.

[10] Jolliffe, I.T. (2002). *Principal component analysis.* Springer series in statistics. Springer, Berlin and New York, 2nd edition.

[11] Hyvärinen, A., Karhunen, J., and Oja, E. (2001) *Independent Component Analysis.* Wiley, New York.

[12] Kawanabe, M., Sugiyama, M., Blanchard, G., Müller, K.-R. (2007). A new algorithm of non-Gaussian component analysis with radial kernel functions. *Ann. Inst. Stat. Math.* **59**, 57–75.

[13] Theis, F.J. and Kawanabe, M. (2007). Uniqueness of non-Gaussian subspace analysis. *Proc. ICA.* **4666**, 917–925, Springer, London.

Многомерные адаптивные регрессионные сплайны*

В. Р. Целых
Celyh@inbox.ru
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В работе рассматриваются многомерные адаптивные регрессионные сплайны. Метод позволяет получить модели, дающие достаточно точную аппроксимацию, даже в тех случаях, когда связи между предикторными и зависимыми переменными имеют немонотонный характер и сложны для приближения параметрическими моделями. Экспериментально исследуется зависимость ошибки аппроксимации от сложности модели. Для иллюстрации работы метода используются тестовые данные, данные ЭКГ и данные из области финансовой математики.

Ключевые слова: *непараметрическая регрессия, многомерные адаптивные регрессионные сплайны, метод наименьших квадратов, обобщенный скользящий контроль.*

Введение

Многомерные адаптивные регрессионные сплайны были впервые предложены Фридманом в 1991 г. [1] для решения регрессионных задач и задач классификации, в которых требуется предсказать значения набора зависимых переменных по набору независимых переменных. Данный метод является непараметрической процедурой, не использующей в своей работе никаких предположений о виде функциональной зависимости между зависимыми и независимыми переменными. MAP-сплайны задаются базисными функциями и набором коэффициентов, полностью определяемых по данным.

MAP-сплайны находят свое применение во многих сферах науки и технологий, например, в предсказании видов распределений по имеющимся данным [2], кишечного поглощения лекарств [3], а также в воспроизведении речи [4] и поиске глобального оптимума в проектировании конструкций [5].

Метод MAP-сплайнов находит искомую зависимость за 2 стадии: “вперед” (forward stage) и “назад” (backward stage) [7]. Первая стадия заключается в добавлении базисных функций к набору, пока не будет достигнут максимальный уровень сложности. На второй стадии из набора удаляются функции, которые вносят наименьший вклад в ошибку.

В данной работе описывается алгоритм построения MAP-сплайнов, тестируется его работа на ряде данных, а также проводится анализ сложности (числа базисных функций) построенной модели.

Описание работы алгоритма

Дана регрессионная выборка:

$$D = \{\mathbf{x}_i, y_i\}_{i=1}^N,$$

где $\mathbf{x}_i \in \mathbb{R}^p, i = 1, \ldots, N$ — векторы независимой переменной, а $y_i,\ i = 1, \ldots, N$ — значения зависимой переменной (непрерывные или бинарные). Связь между y_i и $\mathbf{x}_i\,(i = 1, \ldots, N)$ может быть представлена в виде:

$$y_i = f(x_i^1, x_i^2, \ldots, x_i^p) + \varepsilon = f(\mathbf{x}_i) + \varepsilon,$$

где f — неизвестная функция, а ε — ошибка ($\varepsilon \sim N(0, \sigma^2)$).

Научный руководитель В. В. Стрижов

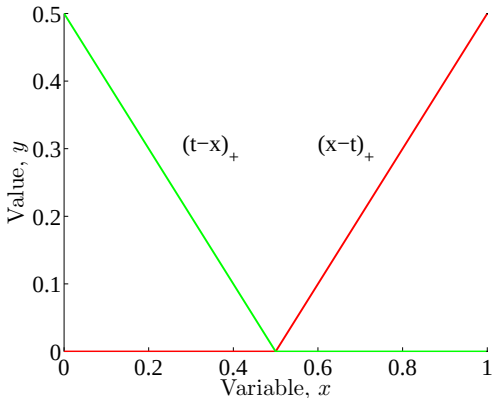


Рис. 1. Базисные функции $(x-t)_+$ и $(t-x)_+$

В одномерном случае MAP-сплайны выражаются через кусочно-линейные базисные функции, $(x-t)_+$ и $(t-x)_+$ с узлом в t. Данные функции являются усеченными линейными функциями (см. рис. 1), при $x \in \mathbb{R}$:

$$(x-t)_+ = \begin{cases} x-t, & \text{если } x > t; \\ 0, & \text{иначе,} \end{cases}$$

$$(t-x)_+ = \begin{cases} t-x, & \text{если } x < t; \\ 0, & \text{иначе.} \end{cases}$$

Эти функции также называются отраженной парой (reflected pair). В многомерном случае для каждой компоненты x^j вектора $\mathbf{x} = (x^1, \ldots, x^j, \ldots, x^p)^T$ строятся отраженные пары с узлами в каждой наблюдаемой переменной x_i^j $(i = 1, 2, \ldots, N; j = 1, 2, \ldots, p)$. Таким образом, набор построенных функций может быть представлен в виде:

$$C := \{(x^j - t)_+, (t - x^j)_+ | \, t \in \{x_1^j, x_2^j, \ldots, x_N^j\}, j \in \{1, 2, \ldots, p\}\}.$$

Если все входные данные различны, то в наборе $2Np$ функций, причем каждая из них зависит только от одной переменной x^j.

Используемые для аппроксимации базисные функции выглядят следующим образом:

$$B_m(\mathbf{x}) = \prod_{k=1}^{K_m} [s_{km} \cdot (x^{v(km)} - t^{km})]_+,$$

где K_m — общее число усеченных линейных функций в m-ой базисной функции, $x^{v(km)}$ — компонента вектора $\mathbf{x}$, относящаяся к k-ой усеченной линейной функции в m-ой базисной функции, t^{km} — соответствующий узел, а $s_{km} \in \{\pm 1\}$.

Построенная модель, как и в линейной регрессии, представляет собой линейную комбинацию, отличие состоит в том, что кроме входных переменных разрешается использовать функции из набора C и их производные функции. Таким образом, модель имеет вид:

$$y = \hat{f}(\mathbf{x}) + \varepsilon = c_0 + \sum_{m=1}^{M} c_m B_m(\mathbf{x}) + \varepsilon,$$

где M — число базисных функций в рассматриваемой модели, а c_0 — общий коэффициент. Как и в линейной регрессии, задав B_m, коэффициенты c_m могут быть найдены по методу наименьших квадратов. Самое главное в данной модели — это выбор базисных функций. В начале модель содержит единственную функцию $B_0(\mathbf{x}) = 1$, а все функции из набора C являются возможными кандидатами для включения в модель.

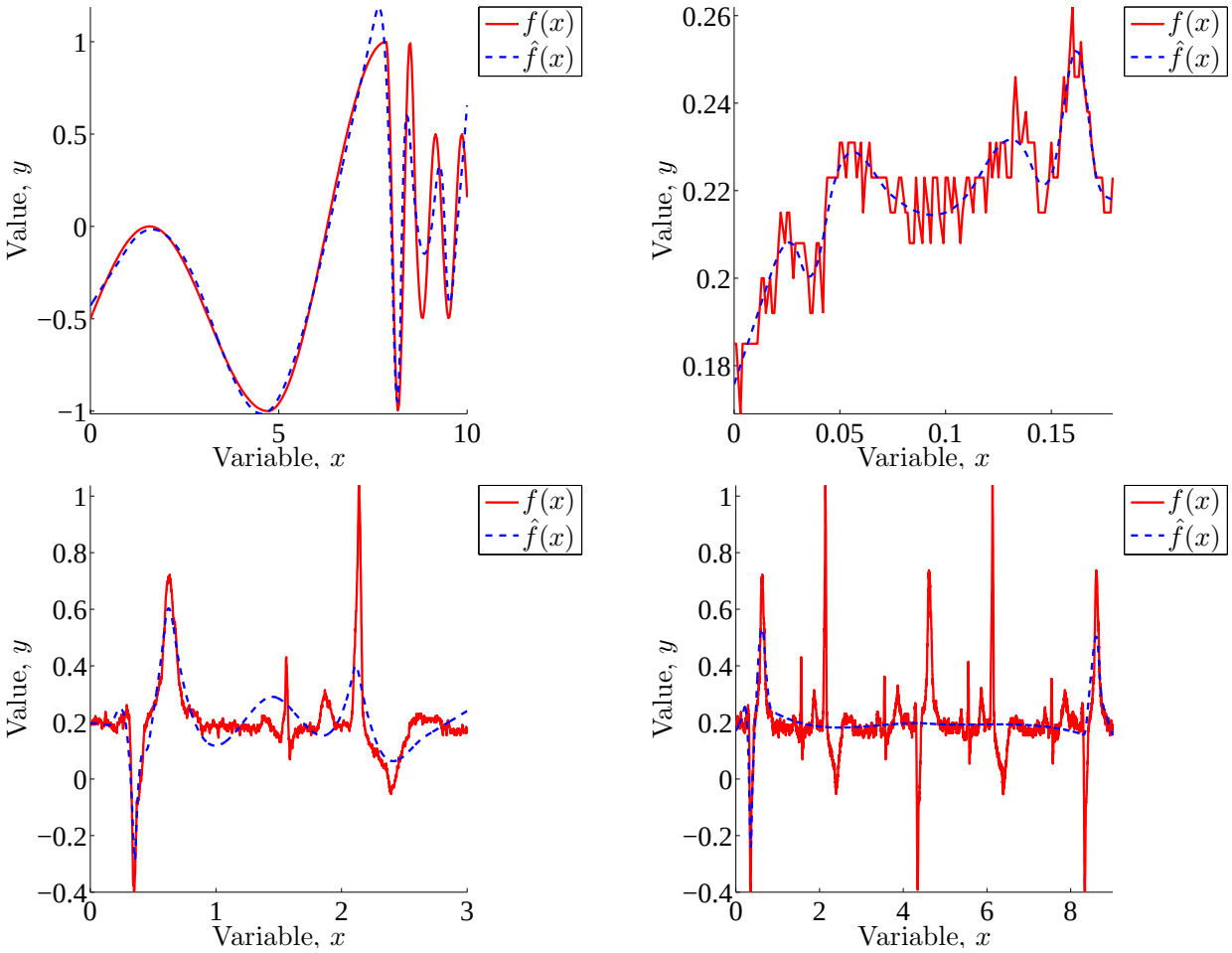


Рис. 2. Функции и построенные аппроксимации

К примеру, следующие функции могут быть базисными:

— 1,
— x^j,
— $(x^j - t^k)_+$,
— $x^l x^j$,
— $(x^j - t^k)_+ x^l$,
— $(x^j - t^k)_+ (x^l - t^h)_+$.

В данном алгоритме каждая базисная функция зависит от разных переменных. Это означает, что $l \neq j$ в базисных функциях, указанных выше. На каждом шаге новая пара базисных функций является произведением функции $B_m(\mathbf{x})$ из множества моделей $\mathcal{M}$ на одну из отраженных пар множества C. Таким образом, в модель $\mathcal{M}$ будет добавлено:

$$\hat{C}_{M+1} B_l(\mathbf{x})(x^j - t)_+ + \hat{C}_{M+2} B_l(\mathbf{x})(t - x^j)_+;$$

что обеспечит наибольшее уменьшение ошибки. Коэффициенты $\hat{C}_{M+1}$ и $\hat{C}_{M+2}$ оцениваются методом наименьших квадратов, как и остальные $M+1$ коэффициентов модели. Процедура добавления функций в модель продолжается до тех пор, пока множество $\mathcal{M}$ содержит менее заданного числа элементов.

Ниже предложены возможные базисные функции:

— $x^j \quad (j = 1, 2, \ldots, p)$,
— $(x^j - t^k)_+$, если x^j уже в модели,
— $x^l x^j$, если x^l и x^j уже в модели,
— $(x^j - t^k)_+ x^l$, если $x^l x^j$ и $(x^j - t^k)_+$ уже в модели,
— $(x^j - t^k)_+ (x^l - t^h)_+$, если $(x^j - t^k)_+ x^l$ и $(x^l - t^h)_+ x^j$ уже в модели.

В конце данной процедуры построена большая модель, которая включает в себя некоторые излишние переменные и обычно чрезмерно подгоняет данные. Необходимо проведение стадии "назад", которая заключается в следующем: на каждом шаге удаляется функция, отсутствие которой вызывает наименьшее увеличение суммы квадратов невязок (RSS). Таким образом, для каждого размера M строится наилучшая модель $\hat{f}_M$. Для оценки оптимальной величины M используется процедура обобщенного скользящего контроля (generalized cross-validation). Данный критерий (также известный как lack-of-fit criterion) выглядит следующим образом [1]:

$$LOF\hat{f}_M = GCV(M) := \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{f}_M(x_i))^2/(1 - C(M)/N)^2,$$

$$C(M) = \mathrm{trace}(\mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T) + 1,$$

где N — число исходных данных, $C(M)$ — оценка штрафов в модели, содержащей M базисных функций, B — матрица размером $M \times N$ ($B_{ij} = B_i(\mathbf{x}_j)$). $C(M)$ — число параметров, подлежащих определению. Один из способов задания числа параметров: $C(M) = r + cK$. Число линейно-независимых базисных функций в модели обозначено r, число узлов, выбранных в стадии "вперед" — K, а число c показывает оценку оптимизации каждой из базисных функций. В общем случае, $c = 3$, но если используемая модель является аддитивной, то $c = 2$. Чем меньше $C(M)$, тем больше получаемая модель и больше число базисных функций, и наоборот соответственно. GCV представляет собой средний квадрат невязок умноженных на коэффициент, характеризующий сложность модели. Таким образом, наилучшая модель состоит из M^* базисных функций, где M^* — решение задачи минимизации $LOF\hat{f}_M$ [8, 9]:

$$M^* = \arg\min_M LOF\hat{f}_M.$$

Особенность метода MAP-сплайнов заключается в использовании кусочно-линейных базисных функций и определенном способе построения модели. Главным свойством кусочно-линейных функций является их способность действовать локально, т. е. принимать ненулевые значения лишь на части их области определения. Результат умножения одной функции на другую отличен от нуля лишь в малой части пространства, где обе функции принимают ненулевые значения. Это и позволяет строить качественные модели, используя сплайны. Если же в качестве базисных функций использовать полиномы, то результат будет хуже по причине того, что полиномы отличны от нуля во всем пространстве.

Вычислительный эксперимент

Для проверки работы алгоритма используется программное обеспечение ARESLab [6]. Сначала тестируется работа алгоритма MAP-сплайнов на простой зависимости $f(x)$, име-

ющей вид:

$$f(x) = \begin{cases} 0.5\sin x - 0.5, & \text{если } 0 \leqslant x < 1.5\pi; \\ \sin x, & \text{если } 1.5\pi \leqslant x < 2.5\pi; \\ -\cos(10x), & \text{если } 2.5\pi \leqslant x < 2.75\pi; \\ 0.5\cos(9x - 0.25\pi), & \text{если } 2.75\pi \leqslant x \leqslant 10; \end{cases}$$

Результат аппроксимации при мощности регрессионной выборки $N = 300$ изображен на рис. 2 в левом верхнем углу. На данном графике красным цветом обозначена исходная зависимость $f(x)$, а синим — ее аппроксимация.

Далее для иллюстрации работы алгоритма рассматривается электрокардиограмма. Результат работы алгоритма при $N = 130$ представлен на рис. 2 в правом верхнем углу. По горизонтальной оси откладывается время t, а по вертикальной — значение напряжения при получении ЭКГ. Красным цветом на графике обозначена исходная зависимость, а синим — ее аппроксимация.

Увеличив мощность регрессионной выборки до 2000, получим аппроксимацию, изображенную на рис. 2 в левом нижнем углу.

И наконец, рассмотрев $N = 6000$, получим аппроксимацию, изображенную на рис. 2 в правом нижнем углу.

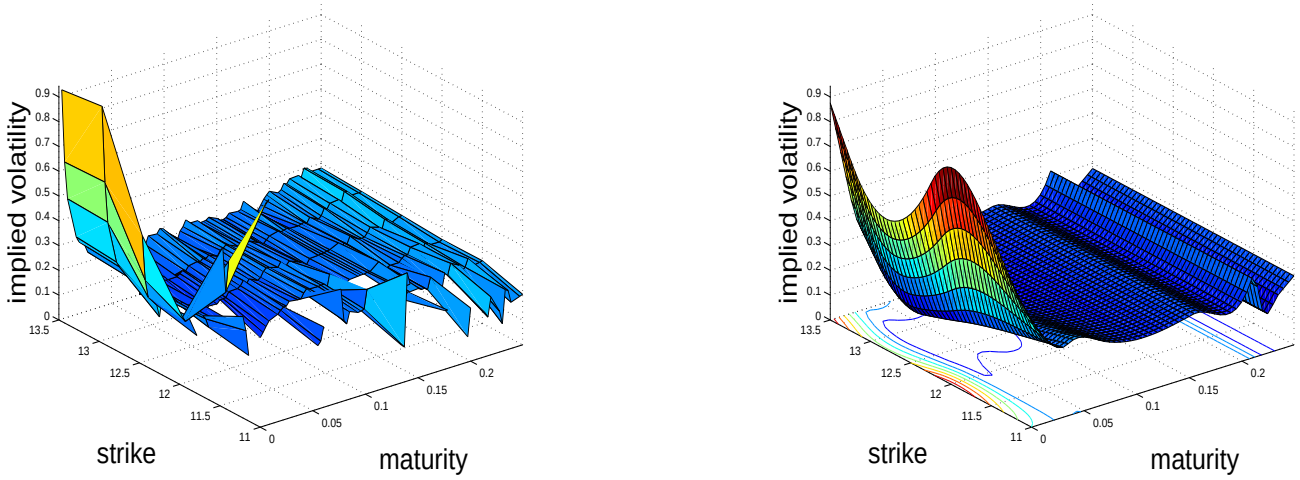


Рис. 3. Функция двух переменных (слева) и построенная аппроксимация (справа)

Данный метод эффективен не только при рассмотрении функций одной переменной, но и в многомерных пространствах. Рассматривается функция двух переменных. Данные взяты из области финансовой математики [10]. По оси x откладывается время до исполнения опциона (maturity), по оси y — цена исполнения опциона (strike), а по оси z — волатильность (implied volatility) опциона [11]. Исходная зависимость представлена на рис. 3 слева. Аппроксимация данной зависимости при максимальном числе пересечений равном 2 и максимальном числе базисных функций на этапе добавления равном 21 изображена на рис. 3 справа.

Из представленных результатов следует, что метод MAP-сплайнов достаточно хорошо описывает любые зависимости.

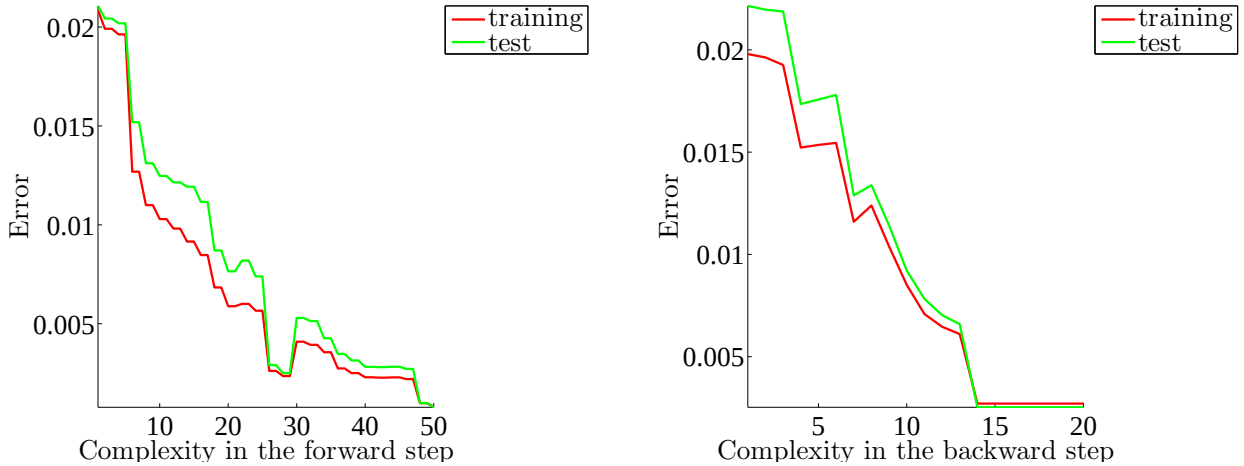


Рис. 4. Зависимость ошибки от числа базисных функций на этапе добавления (слева) и удаления (справа)

Исследуется анализ сложности модели, т. е. число базисных функций в модели. Рассматривается зависимость, которая была приближена на рис. 2 в левом нижнем углу. Выборка случайным образом разделяется на обучающую и проверяющую. Зависимость ошибки (суммы квадратов невязок) при обучении и контроле от числа базисных функций на этапе добавления изображена на рис. 4 слева.

Из данных графиков можно сделать вывод о том, что на стадии "вперед" при числе базисных функций порядка 27 ошибка и на обучающей, и на проверяющей выборке достигает локального минимума.

Зависимость ошибки при обучении и контроле от числа базисных функций на этапе удаления функций из модели (при числе функций равном 27 на этапе добавления) изображена на рис. 4 справа.

Из представленных графиков следует, что на предложенных данных оптимальное число базисных функций на этапе удаления функций из модели равно 14. Значит, во второй стадии эффективного алгоритма из модели удаляется примерно половина базисных функций.

Заключение

В данной работе был описан метод MAP-сплайнов, используемый для нахождения функциональной зависимости между предикторными и зависимыми переменными. Алгоритм был протестирован на ряде данных и показал достаточно хороший результат. Была исследована зависимость ошибки на обучении и контроле от числа базисных функций (как на этапе добавления, так и на этапе удаления функций из модели). При этом оказалось, что число базисных функций в заключительной модели примерно вдвое меньше числа базисных функций в конце первой стадии работы алгоритма.

Литература

[1] Friedman, J.H. *Multivariate adaptive regression splines*, The Annals of Statistics,19, 1 (1991) 1-141.

[2] Elith, J., and Leathwick, J. *Predicting species distribution from museum and herborium records using multiresponse models fitted with multivariate adaptive regression splines*, Diversity and Distributions, 13, 3 (2007) 265-275.

[3] Deconinck, E., Coomons, D., and Heyden, Y.V. *Explorations of linear modeling techniques and their combinations with multivariate adaptive regression splines to predict gastro-intestinal absorption of drugs*, Journal of Pharmaceutical and Biomedical Analysis, 43, 1 (2007) 119-130.

[4] Haas, H., and Kubin, G. *A multi-band nonlinear oscillator model for speech*, Conference Record of the Thirty- Second Asilomar Conference on Signals, Systems and Computers, 1 (1998) 338-342.

[5] Crino, S., and Brown, D.E. *Global optimization with multivariate adaptive regression splines*, IEEE Transactions on Systems Man and Cybernetics Part b — cybernetics, 37, 2 (2007) 333-340.

[6] Gints Jekabsons' webpage, *ARESLab: Adaptive Regression Splines toolbox for Matlab/Octave*, http://www.cs.rtu.lv/jekabsons/regression.html.

[7] Yerlikaya, F. *A new contribution to nonlinear robust regression and classification with MARS and its applications to data mining for quality control in manufacturing*, M.Sc., Department of Scientiic Computing (2008) 1-102.

[8] Di, W. *Long Term Fixed Mortgage Rate Prediction Using Multivariate Adaptive Regression Splines*, School of Computer Engineering, Nanyang Technological University, 2006.

[9] Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning, Data Mining, Inference and Prediction*, Springer, 2001.

[10] Strijov, V., *Volatility smile modelling: two-dimensional linear regression demo*, http://strijov.com/sources/demo_linfit_options.php#1.

[11] Стрижов, В., и Сологуб, Р. *Индуктивное построение регрессионных моделей волатильности опционных торгов*, Вычислительные технологии, том 14, №5, 2009.

Выбор признаков и шаговая логистическая регрессия для задачи кредитного скоринга*

А. А. Адуенко
aduenko1@gmail.com
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

Работа посвящена выбору оптимального набора признаков для определения уровня надежности заемщиков, подавших заявку на банковский кредит. Для ответа на поставленный вопрос оценивается вероятность невозврата кредита. Для отбора признаков используется шаговая регрессия, исследуется зависимость информативности отобранных признаков от параметров шаговой регрессии. В вычислительном эксперименте алгоритм тестируется на данных потребителей, подававших заявки на кредиты в определенный банк, а также на данных об отклике клиентов на маркетинговую кампанию банка.

Ключевые слова: *банковский кредит, логистическая регрессия, выбор признаков, функция эмпирического риска, вероятность невозврата.*

Введение

В работе рассматривается задача кредитного скоринга [1]. По данным заемщиком ответам на фиксированный набор вопросов анкеты требуется определить, в состоянии ли тот вернуть кредит банку. Кроме того, одной из основных задач является выделение некоторого небольшого набора признаков, по которому наиболее точно можно будет судить о кредитоспособности. Основным источником алгоритмов и способов отбора признаков служили [1, 2]. Модифицированные версии алгоритмов, представленных там и применяются в работе. Они основаны на подсчёте WOE (англ. weight of evidence), меры информативности соответствующего значения признака, для каждого из признаков в отдельности. Для поиска весов признаков из найденного оптимального набора используется логистическая регрессия [1, 2, 3, 4], а для их отбора— шаговая логистическая регрессия [1, 2]. Для контроля качества на тестовой выборке рассчитывается функция эмпирического риска [5, 6]. В вычислительном эксперименте представлены результаты работы построенного алгоритма на данных об отклике клиентов на маркетинговую кампанию банка [7]. Также рассмотрены свойства алгоритма при работе с данными анкет по потребительским кредитам [8].

Постановка задачи

Имеются исходные данные – выборка $D = \{(x_i, y_i)\}$, $i \in \mathcal{I} = \mathcal{S} \sqcup \mathcal{T}$: матрица признаков $X \in \mathbb{R}^{m \times n}$ (m–число записей данных, а n–количество признаков) и вектор ответов $\mathbf{y}$, $y_i \in \{-1, 1\}$. Здесь -1 означает, что заемщик кредит вернул (класс Y_{-1}), а 1— не вернул (класс Y_1). Разбиение на обучающую выборку $S\{(x_i, y_i)\}$, $i \in \mathcal{S}$ и тестовую $T\{(x_i, y_i)\}$, $i \in \mathcal{T}$ осуществляется случайно. Предполагается, что $x_{ij} \in \mathbb{Z}$.

Для определения уровня кредитоспособности заемщиков используется модель логистической регрессии

$$f(\mathbf{x}, \mathbf{w}) = \frac{1}{1 + \exp\left(-\langle \mathbf{x}, \mathbf{w} \rangle\right)}. \tag{1}$$

Научный руководитель В. В. Стрижов

Здесь $\mathbf{w} \in \mathcal{W} = \mathbb{R}^n$ вектор параметров модели, а $\mathbf{x} \in \mathbb{Z}^n$ — вектор значений признаков объекта. $f(\mathbf{x}, \mathbf{w})$ задает оценочную вероятность того, что рассматриваемый объект принадлежит классу Y_{-1}.

Требуется по обучающей выборке S оценить параметр $\mathbf{w}^*$ модели (1), чтобы далее классифицировать объекты в предположении, что из исходного множества признаков $\{\boldsymbol{\chi}_j\}$, $j \in \mathcal{J} = \{1, \ldots, n\}$ отобрано некоторое подмножество $\{\boldsymbol{\chi}_j\}$, $j \in \mathcal{A}$ оптимальных согласно (3) признаков, $|\mathcal{A}| = n^* \leqslant n$. Параметр находится путем максимизации качества модели на обучающей выборке S.

В качестве меры качества используется функция эмпирического риска

$$R(\mathbf{w}, \mathcal{X}, \mathcal{A}) = \sum_{i=1}^{|\mathcal{X}|} \ln\left(1 + \exp\left(-y_i \cdot \langle \mathbf{x}_i, \mathbf{w}\rangle\right)\right), \tag{2}$$

где $\mathcal{X}$ — некоторая выборка объектов, $(\mathbf{x}_i,\ y_i) \in \mathcal{X}$, y_i задает класс объекта $\mathbf{x}_i$. $\mathcal{A}$ —набор индексов используемых признаков. Поиск оптимального набора параметров в соответствии с (2) осуществляется следующим образом:

$$\mathbf{w}^* = \underset{\mathbf{w} \in \mathcal{W} = \mathbb{R}^n}{\arg\min} R(\mathbf{w}, S, \mathcal{A}) \tag{3}$$

Задачу поиска оптимального набора признаков $\{\boldsymbol{\chi}_j\}$, $j \in \mathcal{A}$ можно записать в виде

$$\mathcal{A} = \underset{\mathbf{w} \in \mathcal{W} = \mathbb{R}^n, \mathcal{A} \subseteq \mathcal{J}}{\arg\min} R(\mathbf{w}, \mathcal{X}, \mathcal{A}) \tag{4}$$

Задача нахождения оптимального набора признаков решается в работе с помощью шаговой логистической регрессии.

Нахождение весов признаков

Перейдем к нахождению весов признаков. Эта задача является одной из основных, поскольку от того, с каким весом тот или иной признак $\boldsymbol{\chi}_j$ войдет в модель, существенно зависит поведение классификатора (7).

Присоединим каждому вектору $\mathbf{x}_i$ в качестве первого элемента -1. Заменим $\mathcal{A}$ на $\mathcal{A} \cup \{1\}$, сохраняя обозначение $\mathcal{A}$. При исключении из $\mathbf{x}_i = [x_{i1}, \ldots, x_{in}]$ элементов x_{ij}, $j \notin \mathcal{A}$ сохраним для полученного вектора обозначение $\mathbf{x}_i$.

Для оценки $\mathbf{w}^*$, вектора параметров модели (1), пользуясь формулой (2), запишем производную функции эмпирического риска:

$$\frac{\partial R(\mathbf{w}, \mathcal{X}, \mathcal{A})}{\partial \mathbf{w}} = -\sum_{i=1}^{|\mathcal{X}|} \frac{\exp\left(-\mathbf{y}_i \cdot \langle \mathbf{x}_i, \mathbf{w}\rangle\right)\mathbf{y}_i}{1 + \exp\left(-\mathbf{y}_i \cdot \langle \mathbf{x}_i, \mathbf{w}\rangle\right)} \cdot \mathbf{x}_i. \tag{5}$$

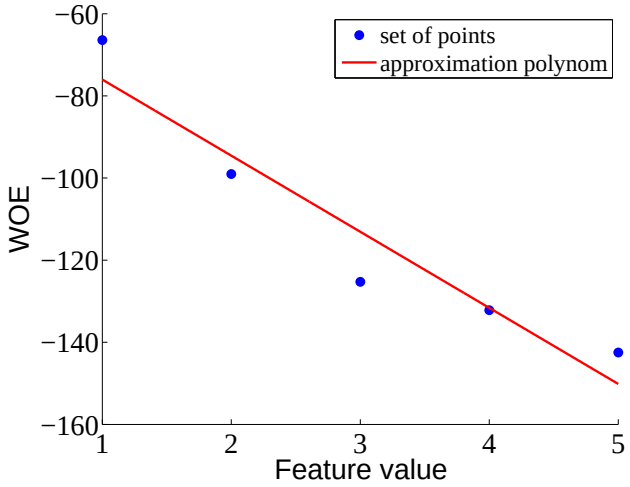
Минимум эмпирического риска достигается в точке $\mathbf{w}^*$, определяемой из соотношения:

$$\frac{\partial R(\mathbf{w}^*, \mathcal{X}, \mathcal{A})}{\partial \mathbf{w}} = \mathbf{0} \tag{6}$$

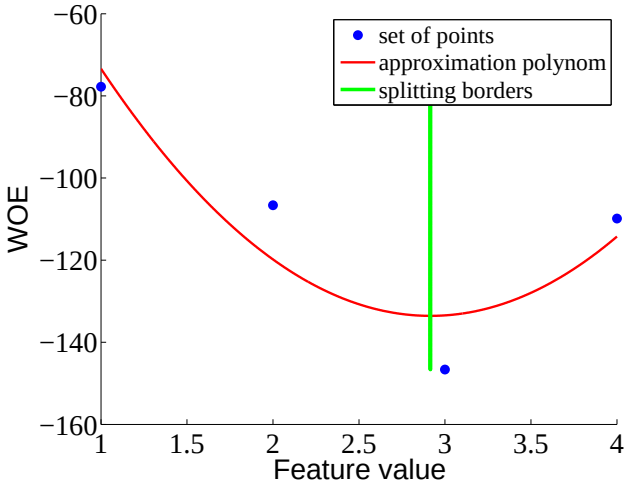
Точку, удовлетворяющую (6), найдем методом градиентного спуска [9].

По полученным весам $\mathbf{w}^*$ строим классификатор:

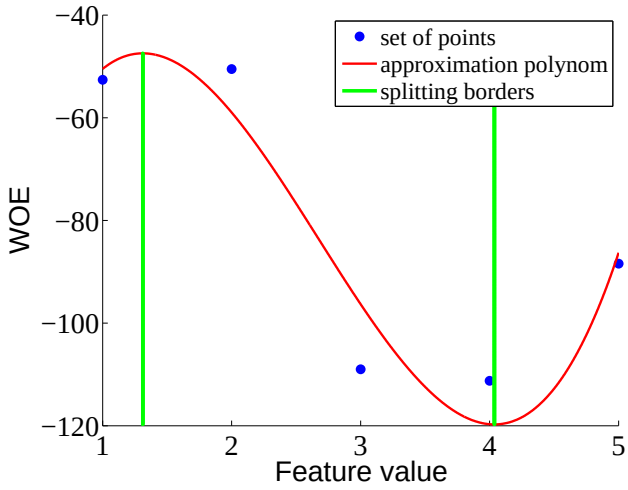
$$\psi(\mathbf{x}_i) = \operatorname{sign} \langle \mathbf{w}^*, \mathbf{x}_i \rangle, \tag{7}$$



(a) Число монотонных компонент $L = 1$



(b) Число монотонных компонент $L = 2$



(c) Число монотонных компонент $L = 3$

Рис. 1. Приближение полиномом зависимости WOE от значения признаков и иллюстрация порождения признаков

где $\mathbf{x}_i$ произвольный объект. Вероятности попасть в соответствующие классы определяется сигмоидной функцией:

$$P(Y_{-1}|\mathbf{x}_i) = f(\mathbf{x}_i, \mathbf{w}^*) = \frac{1}{1 + \exp\left(- < \mathbf{w}^*, \mathbf{x}_i >\right)}, \tag{8}$$

$$P(Y_1|\mathbf{x}_i) = 1 - f(\mathbf{x}_i, \mathbf{w}^*) = \frac{\exp\left(- < \mathbf{w}^*, \mathbf{x}_i >\right)}{1 + \exp\left(- < \mathbf{w}^*, \mathbf{x}_i >\right)}. \tag{9}$$

Теперь вернемся к отбору и порождению признаков.

Порождение признаков

Рассмотрим подробнее процесс порождения признаков. Порождение признаков необходимо, поскольку возможно часть исходных признаков мало информативна для линейного классификатора (7). Примером такого признака может служить возраст. Банковская статистика подтверждает, что лучше всего кредиты возвращают люди среднего возраста, в то время как молодые и пожилые чаще их не возвращают. Ясно, что никакой вес $\mathbf{w}_{age}$

возраста не позволит учесть эту особенность, поскольку классификатор линейный (7). Проведем процесс порождения признаков так, чтобы избежать подобных проблем.

Пусть $\boldsymbol{\chi}_j$ произвольный признак. Пусть его значения в порядке возрастания на обучающей выборке S есть $v_1, \ldots, v_k$, $v_i < v_j \, \forall i, j : i < j$. Определим для каждого значения рассматриваемого признака WOE (англ. weight of evidence) по формуле:

$$WOE_j(v_q) = \log \frac{[y_i = -1 \,\&\, \boldsymbol{\chi}_j(\mathbf{x}_i) = v_q] + 1}{[y_i = 1 \,\&\, \boldsymbol{\chi}_j(\mathbf{x}_i) = v_q] + 1}, \tag{10}$$

где $(\mathbf{x}_i, y_i) \in S$. [условие]—количество элементов выборки, на которых условие выполнено.

Пусть $e_q = WOE_j(v_q) \, \forall q \in \{1, \ldots, k\}$. Пусть также $\{e_1, \ldots, e_{i_1}\}$, $\{e_{i_1+1}, \ldots, e_{i_2}\}, \ldots$, $\{e_{i_{L-1}+1}, \ldots, e_{i_L} = e_k\}$ есть монотонные последовательности. Причем соседние последовательности обладают противоположным направлением роста. Назовем L числом монотонных компонент. Так как построенный классификатор линейный, то можно предположить, что наивысшее качество классификации в терминах (2) и (4) наблюдается по признакам, у которых L мало, а лучше $L = 1$.

Рассматриваемый признак $\boldsymbol{\chi}_j = [x_{1j}, \ldots, x_{Mj}]$, $M = |S|$, $x_{qj} \in \mathbb{Z}$. Разобьем числовую ось $\mathbb{R}$ на несколько полуинтервалов, а именно на L. Для этого определим $L - 1$ место разбиения $\mathbb{R}$: $d_1 < \ldots < d_{L-1}$. Положим также, что $d_0 = -\infty$, а $d_l = \infty$.

Заменяем исходный вектор признаков $\boldsymbol{\chi}_j$ на L новых: $\boldsymbol{\chi}_j^1, \ldots, \boldsymbol{\chi}_j^L$ по такому правилу: если для объекта $\mathbf{x}_q$ выборки D $\boldsymbol{\chi}_j(\mathbf{x}_q) = v_c$ и $v_c \in (d_i, d_{i+1}]$ для некоторого $i \in \{0 \ldots L-1\}$, то $\boldsymbol{\chi}_j^s(\mathbf{x}_q) = 0 \, \forall s \neq i+1$ и $\boldsymbol{\chi}_j^{i+1}(\mathbf{x}_q) = \boldsymbol{\chi}_j(\mathbf{x}_q)$.

Теперь определим как найти $d_1, \ldots, d_{L-1}$. Для этого найдем полином степени L, наименее уклоняющийся в среднеквадратическом от точек $\{v_i, e_i\}_{i=1}^k$, то есть

$$\mathbf{c}^* = \arg\min \|A\mathbf{c} - \mathbf{e}\|^2, \tag{11}$$

где матрица A имеет следующий вид

$$\begin{pmatrix} a_{11} & \ldots & a_{1L} \\ \vdots & \ddots & \vdots \\ a_{M1} & \ldots & a_{ML} \end{pmatrix},$$

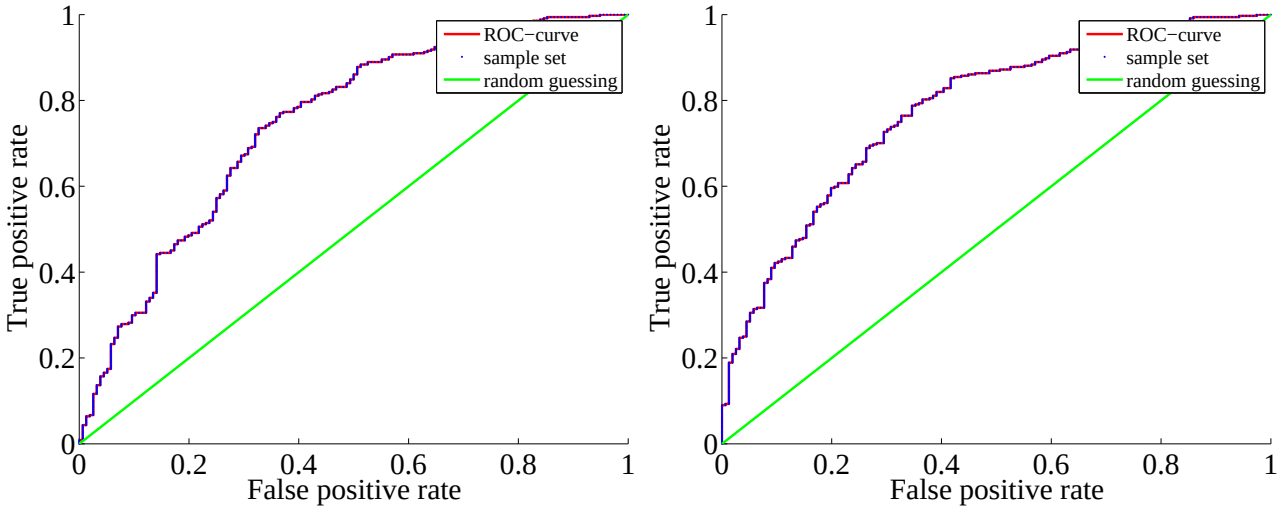
где $a_{kl} = (v_k)^l$. Осталось найти набор нулей B производной полученного многочлена с коэффициентами $\mathbf{c}^*$, определяемыми из (11). Именно нули производной и задают границы полуинтервалов $d_1, \ldots, d_{L-1}$. На рис.1 приведены примеры разбиений действительной оси $\mathbb{R}$ для признаков с разным числом монотонных компонент L.

После выполнения процедуры для каждого признака получаем новую матрицу плана $\mathcal{X}$. Далее применяем алгоритм отбора признаков.

Отбор признаков

Зачастую то, вернет или не вернет заемщик кредит, заметно зависит не от всего набора признаков, а лишь от их части. Для удаления из множества признаков $\boldsymbol{\chi}_j$, $j \in \mathcal{J}$ таких неинформативных признаков и применим их отбор.

Для отбора признаков воспользуемся шаговой логистической регрессией. Подробно алгоритм шаговой логистической регрессии изложен в [2], в работе же опишем лишь общую его идею.



(a) Исходные признаки (b) Разделенные признаки

Рис. 2. ROC-кривые для исходных и разделенных признаков

Для поиска оптимального набора признаков будем пользоваться следующим жадным алгоритмом: изначально имеем модель f, в которой ровно один признак: $\mathcal{A} = \{1\}$. Первый признак есть константа, а именно: $\boldsymbol{\chi}_1(\mathbf{x}_i) = -1\,\forall \mathbf{x}_i$.

Этот признак всегда будет в модели. Все дальнейшие шаги относятся ко всем признакам, кроме этого.

На каждом следующем шаге проверяем сначала возможность добавить новый признак в модель f, а затем возможность удалить в соответствии со следующими правилами.

Добавление признака

Пусть в модели f уже есть признаки $\boldsymbol{\chi}_{j_1}, \ldots, \boldsymbol{\chi}_{j_k}$, $j_1 = 1$, то есть $\mathcal{A} = \{j_1, \ldots, j_k\}$. Пусть также признаки $\boldsymbol{\chi}_{j_{k+1}}, \ldots, \boldsymbol{\chi}_{j_n}$ не находятся в модели f. Обозначим $\tilde{\mathbf{w}}\,(\mathcal{A})$ значение вектора весов признаков $\{\boldsymbol{\chi}_j\}$, $j \in \mathcal{A}$, определяемое из (6).

Эмпирический риск для модели f в соответствии с (2) обозначим $R_0 = R(\tilde{\mathbf{w}}(\mathcal{A}), S, \mathcal{A})$. Для каждого $s \in \{k+1, \ldots, n\}$ обозначим $\mathcal{A}' = \mathcal{A} \cup \{j_s\}$. Пусть эмпирический риск получаемой модели f' для каждого $s \in \{k+1, \ldots, n\}$ есть R_s. Среди всех R_s выбираем наименьший:

$$s^* = \underset{s \in \{k+1,\ldots,n\}}{\arg\min} R_s$$

Это же в терминах (2) и введенного $\tilde{\mathbf{w}}\,(\mathcal{A})$ выглядит следующим образом:

$$j_{s^*} = \underset{j_s \in \mathcal{J} \setminus \mathcal{A}}{\arg\min} R(\tilde{\mathbf{w}}(\mathcal{A}'), S, \mathcal{A}'). \tag{12}$$

Обозначим $j_{s^*} = j^*$. Далее считаем рисковую разницу G_{j^*} и вероятность p, отражающую значимость признака

$$G_{j^*} = 2 \cdot (R_0 - R_{s^*})\,, \tag{13}$$

$$p = Pr\left[\chi^2(\nu) > G_{j^*}\right]. \tag{14}$$

Здесь $\chi^2(n)$ имеет функцию плотности вероятности

$$f_{\chi^2}(x) = \begin{cases} 0, & \text{если } x \leqslant 0; \\ \dfrac{x^{\frac{n}{2}-1} \cdot \exp\left(-\frac{x}{2}\right)}{2^{\frac{n}{2}} \cdot \Gamma\left(\frac{n}{2}\right)}, & \text{если } x > 0. \end{cases}$$

Для дискретного признака в качестве ν используем число разных значений признака. Если признак непрерывный, полагаем $\nu = 1$. При этом чем больше значение p для признака, тем менее признак значимый. Поэтому установим границу отсечения P_E. Если для признака $\boldsymbol{\chi}_{j^*}$ $p < P_E$, этот признак $\boldsymbol{\chi}_{j^*}$ добавляется в модель f, то есть $\mathcal{A} \to \mathcal{A} \cup \{j^*\}$, иначе модель f остается неизменной, то есть $\mathcal{A}$ не изменяется.

Удаление признака

После выполнения первой части шага проверяем в полученной модели все признаки на значимость. Пусть в модели f уже есть признаки $\boldsymbol{\chi}_{j_1}, \ldots, \boldsymbol{\chi}_{j_{k'}}$, $j_1 = 1$, то есть $\mathcal{A} = \{j_1, \ldots, j_{k'}\}$, где $k' = k$, если на шаге добавления признаки не добавлялись и $k' = k + 1$ иначе. Пусть также признаки $\boldsymbol{\chi}_{j_{k'+1}}, \ldots, \boldsymbol{\chi}_{j_n}$ не находятся в модели f.

Эмпирический риск для модели f в соответствии с (2) обозначим $R_0 = R(\tilde{\mathbf{w}}(\mathcal{A}), S, \mathcal{A})$. Для каждого $s \in \{1, \ldots, k'\}$ обозначим $\mathcal{A}' = \mathcal{A} \backslash \{j_s\}$. Пусть эмпирический риск получаемой модели f' для каждого $s \in \{1, \ldots, k'\}$ есть R_s. Среди всех R_s выбираем наименьший:

$$s^* = \underset{s \in \{1, \ldots, k'\}}{\arg\min} R_s$$

Это же в терминах (2) и введенного $\tilde{\mathbf{w}}(\mathcal{A})$ выглядит следующим образом:

$$j_{s^*} = \underset{j_s \in \mathcal{A}}{\arg\min} R(\tilde{\mathbf{w}}(\mathcal{A}'), S, \mathcal{A}'). \tag{15}$$

Обозначим $j_{s^*} = j^*$. Далее считаем рисковую разницу G_{j^*} и вероятность p, отражающую значимость признака

$$G_{j^*} = 2 \cdot (R_0 - R_{s^*}), \tag{16}$$

$$p = Pr\left[\chi^2(\nu) > G_{j^*}\right]. \tag{17}$$

Руководствуясь значением вероятности p как оценкой значимости признака, устанавливаем границу отсечения $P_R > P_E$. Если окажется, что $p > P_E$, то признак $\boldsymbol{\chi}_{j^*}$ удаляется из модели, то есть $\mathcal{A} \to \mathcal{A} \backslash \{j^*\}$. Иначе модель не изменяется, то есть $\mathcal{A}$ остается прежним.

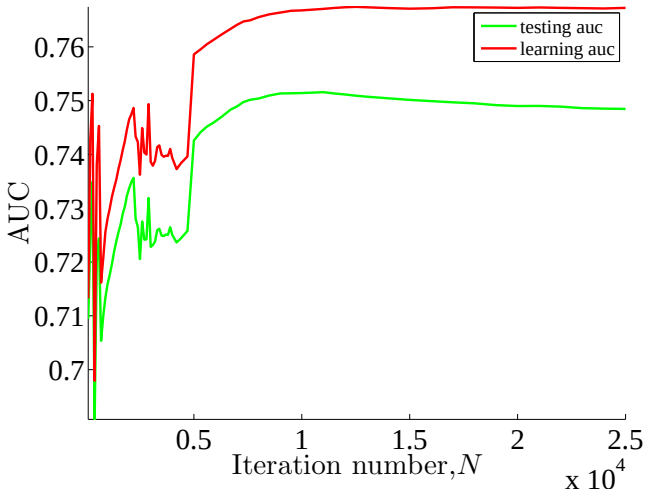
Переход на следующий шаг происходит, если было совершено или удаление, или добавление, иначе алгоритм заканчивает свою работу. По окончании работы алгоритма получаем некоторый набор признаков, по которому можно классифицировать объекты.

Вычислительный эксперимент

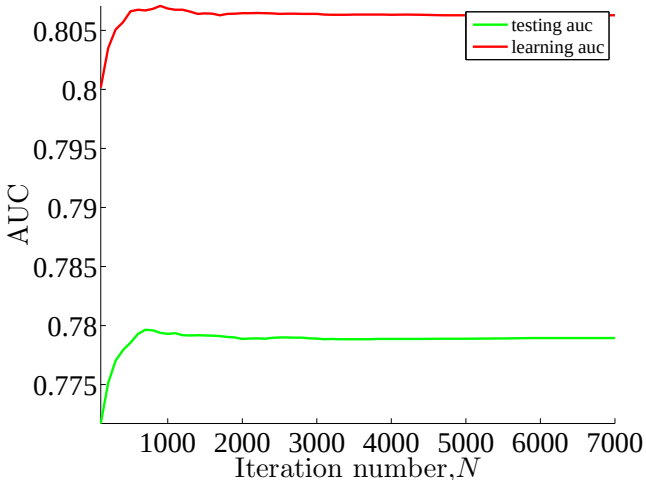
В вычислительном эксперименте продемонстрируем работу приведенных алгоритмов на следующих данных:

— Данные анкет по потребительским кредитам [8] (1000 объектов, 24 признака)

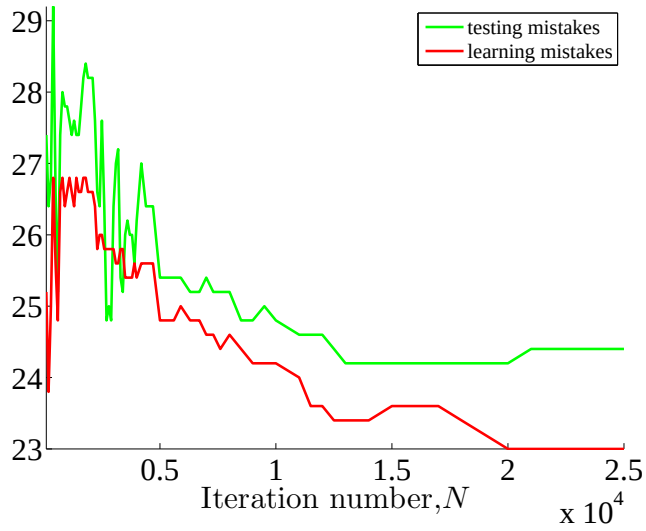
— Данные отклика клиентов на маркетинговую кампанию ОТР-банка [7] (15223 объекта, 36 признаков)



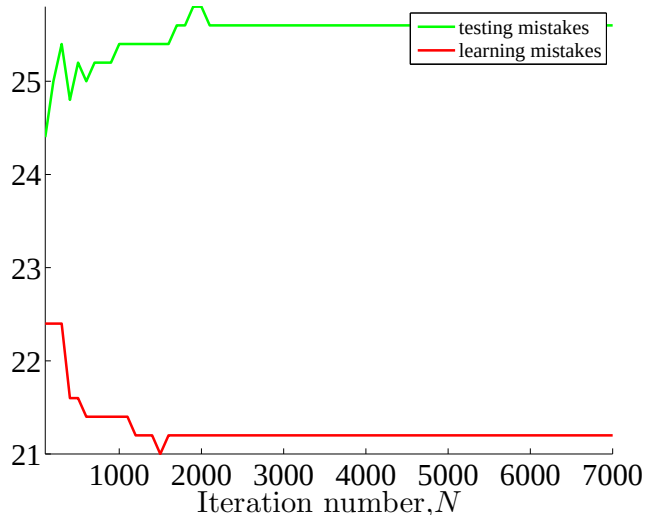
(a) Исходные признаки



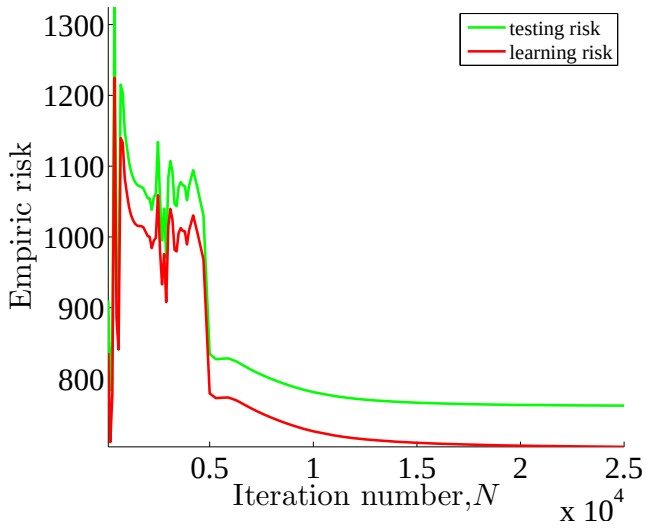
(b) Разделенные признаки



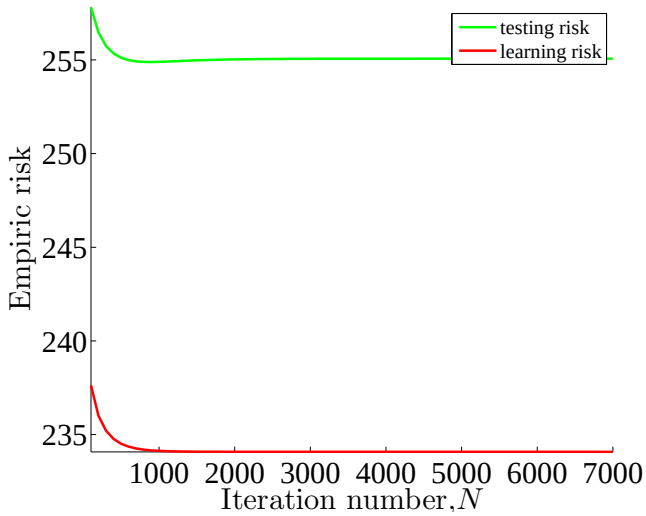
(c) Исходные признаки



(d) Разделенные признаки

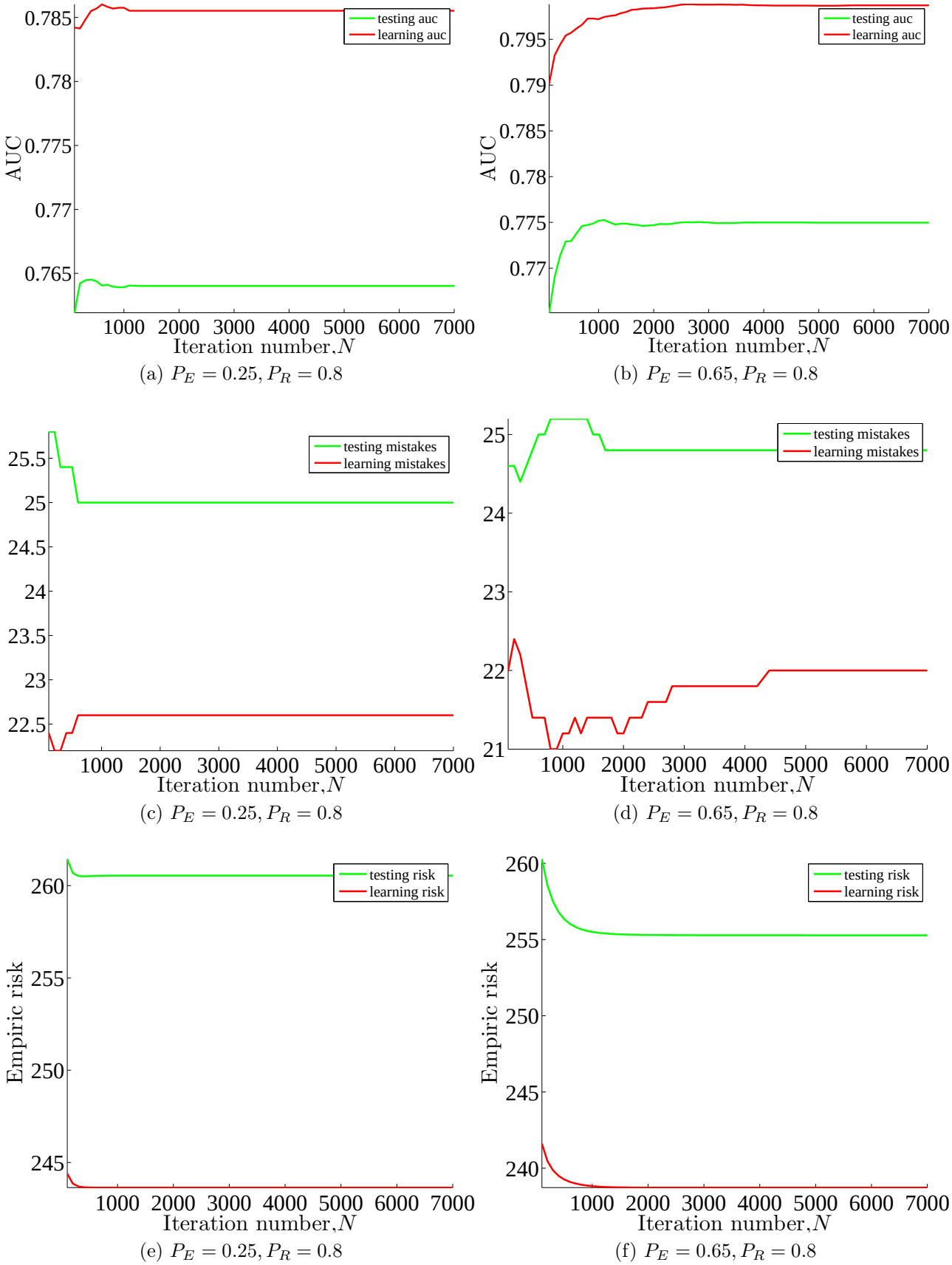


(e) Исходные признаки



(f) Разделенные признаки

Рис. 3. Зависимость площади AUC под ROC-кривой, процента ошибок и эмпирического риска для исходных и разделенных признаков от числа итераций метода градиентного спуска

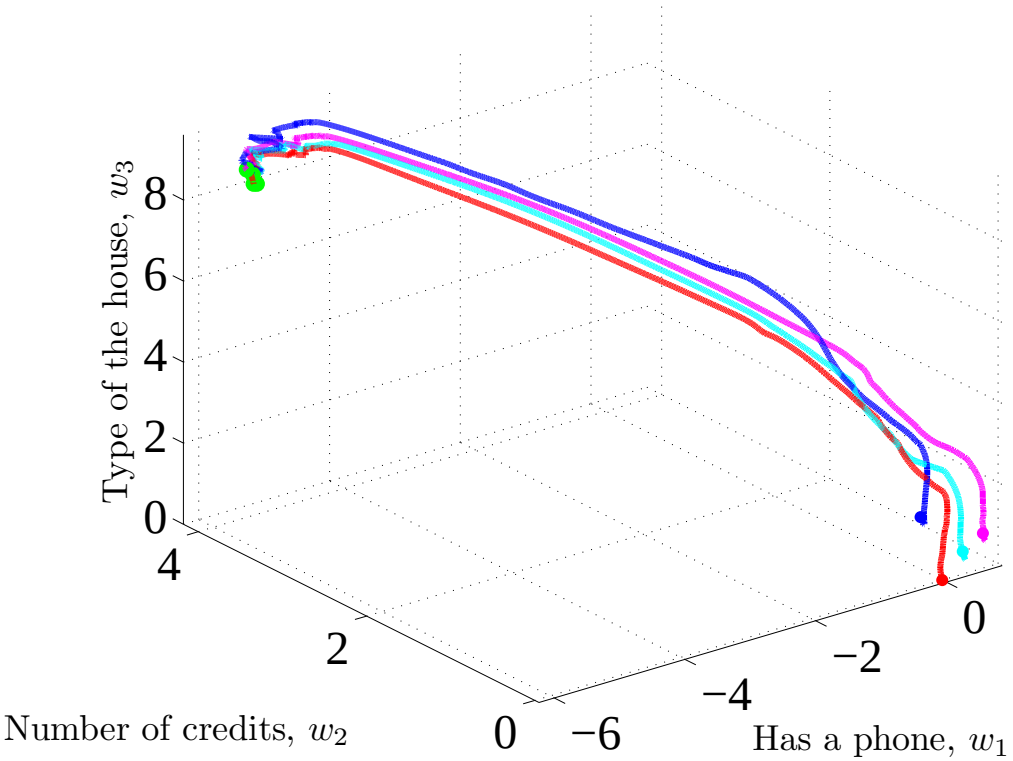


(a) $P_E = 0.25, P_R = 0.8$ (b) $P_E = 0.65, P_R = 0.8$

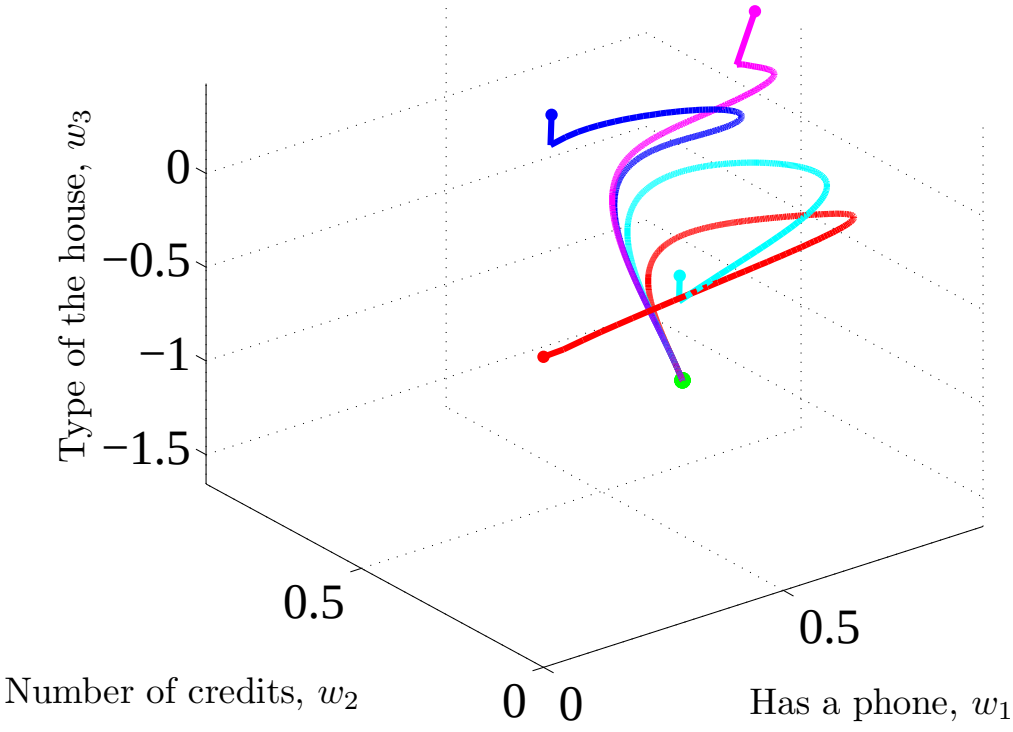
(c) $P_E = 0.25, P_R = 0.8$ (d) $P_E = 0.65, P_R = 0.8$

(e) $P_E = 0.25, P_R = 0.8$ (f) $P_E = 0.65, P_R = 0.8$

Рис. 4. Зависимость площади под *ROC*—кривой, процента ошибок и эмпирического риска от числа итераций градиентного спуска для $P_E = 0.25$, $P_R = 0.8$ и $P_E = 0.65$, $P_R = 0.8$

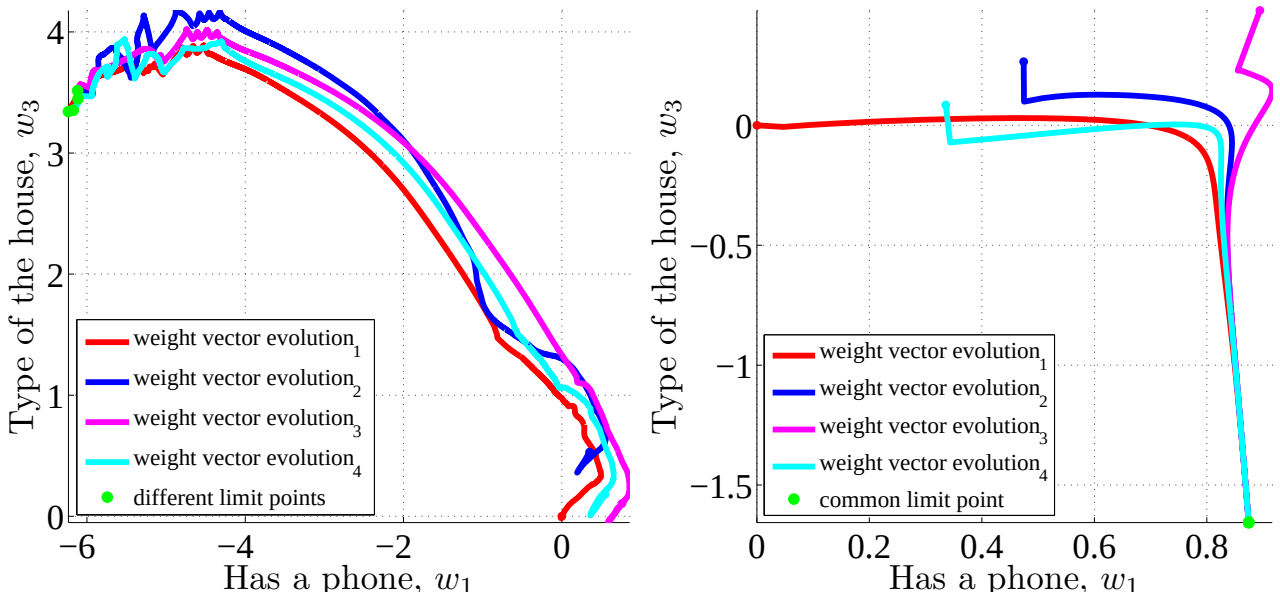


(a) Исходные признаки



(b) Отобранные признаки, $P_E = 0.65$, $P_R = 0.8$

Рис. 5. Сходимость к оптимальному вектору весов для исходных и отобранных признаков из двух разных начальных приближений в пространстве трех признаков



(a) Исходные признаки (b) Отобранные признаки, $P_E = 0.65$, $P_R = 0.8$

Рис. 6. Сходимость к оптимальному вектору весов для исходных и отобранных признаков из двух разных начальных приближений в плоскости двух признаков

В данных по потребительским кредитам пропусков и данных в текстовом виде не было, поэтому эти данные использовались без обработки. В данных же OTP-банка были пропуски, а также признаки в текстовом виде. Пропуски были заполнены нулями, а признаки в текстовом виде не учитывались.

Целью вычислительного эксперимента было проверить предположения о том, что построенные алгоритмы порождения и отбора признаков позволяют повысить качество классификации в терминах (2), выделить значимые признаки (4) и тем самым сократить объем обрабатываемой информации. Также требовалось оценить как применение алгоритмов сказывается на требуемых для обработки вычислительных ресурсах.

Иллюстрация порождения признаков

Проиллюстрируем описанное выше на реальных данных о потребительских кредитах [8]. Начнем с примеров признаков, обладающих разным числом монотонных компонент L, выделяемых в отдельные признаки. Приведем иллюстрации приближения полиномом (11) с коэффициентами $\mathbf{c}^*$ зависимости $WOE_j(v_q)$ для трех разных признаков. На графиках на рис. 1 также изобразим найденные границы разбиения действительной оси $\mathbb{R}$ $d_1, \ldots, d_{L-1}$ как нулей производной полученного полинома с коээфициентами $\mathbf{c}^*$. При этом на рис. 1(а) границы разбиения не показаны, так как для этого признака $L = 1$. Точки на графиках соответствуют всем представленным в обучающей выборке S значениям рассматриваемого признака $\boldsymbol{\chi}_j$ $v_1, \ldots, v_K$ и посчитанным для них по обучающей выборке S в соответствии с (10) значениям $WOE_j(v_q)$, $q \in \{1, \ldots, K\}$. Линией на графиках рис. 1 показан полином степени L с коэффициентами $\mathbf{c}^*$, наименее уклоняющийся в среднеквадратическом от этих точек.

Теперь построим логистическую регрессию на исходных признаках. Приводим *ROC*-кривые (на рис. 2) [5], график площади под ними *AUC*, график процента ошибок и график эмпирического риска (рис. 3) в зависимости от числа итераций алгоритма градиентного спуска, то есть фактически от того, насколько точно найден оптимальный для обучающей

выборки T вектор весов признаков $\mathbf{w}^*$. Прямой линией на рис. 2 показан худший классификатор, основанный на случайном угадывании. На рис. 2 и рис. 3 слева приведены графики для исходных признаков, а справа – для разделенных.

Графики на рис. 2 и рис. 3 иллюстрируют тот факт, что в случае разделенных признаков наступает заметно более ранняя сходимость. Возможно, это объясняется оптимальной структурой построенных признаков для линейного классификатора. Более того, эксперимент показывает, что вектор весов $\mathbf{w}$ в случае с неразделенными признаками очень быстро растет по норме, а при регуляризации эмпирического риска падает эффективность. Напротив, $\|\mathbf{w}\|$ слабо растет на разделенных признаках.

Отбор признаков и сравнение результатов

В качестве данных будем использовать данные клиентов, подававших заявки на потребительские кредиты, а также данные об отклике людей на маркетинговую компанию банка.

Так как людей, подающих заявки на кредиты в банк обычно заметно больше, чем компаний, особенно важно выбрать некоторый небольшой набор признаков для идентификации надежного заемщика, чтобы работать с меньшими массивами данных.

Именно для выделения наиболее информативных признаков и будет использоваться описанный алгоритм шаговой логистической регрессии.

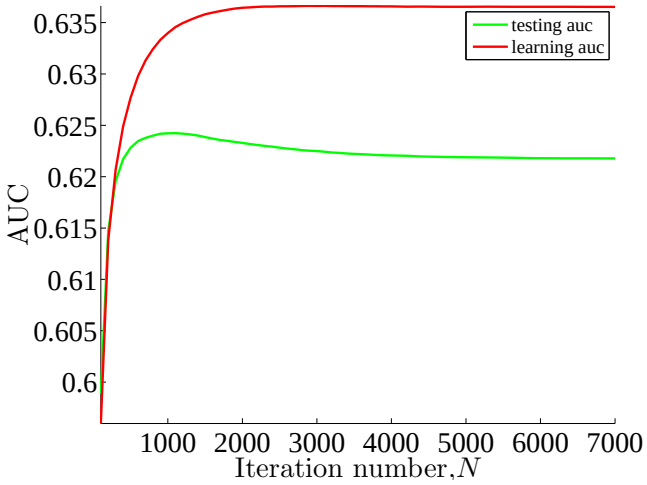
Для реализации алгоритма требуется задать границ отсечения для шагов добавления и удаления признаков P_E и P_R. В зависимости от выбора P_E и P_R алгоритм будет выделять в общем случае разные признаки и разное количество таковых. На опыте оказалось, что рекомендованное в [2] значение $P_E = 0.15 - 0.25$ для рассматриваемых задач слишком мало, а также, что от P_R почти ничего не зависит.

При указанном значении $P_E = 0.25$ алгоритм отбирает очень узкий набор признаков, которого не вполне хватает. Приведём графики зависимости площади под ROC-кривой, эмпирического риска и процента ошибок от числа итераций алгоритма градиентного спуска для следующих значений P_E и P_R: 0.25 и 0.8 (рекомендация [2]), 0.5 и 0.8 (для маркетинговой компании), 0.65 и 0.8 (для потребительских кредитов). Для потребительских кредитов графики приведены на рис. 4, для маркетинговой кампании — на рис. 7. На этих рисунках (рис.4 и рис. 7) слева приведены графики для $P_E = 0.25$, $P_R = 0.8$, а справа — для $P_E = 0.65$, $P_R = 0.8$ и $P_E = 0.5$, $P_R = 0.8$ соответственно. Две последних пары значений оптимальны для соответствующих задач как показывает эксперимент.

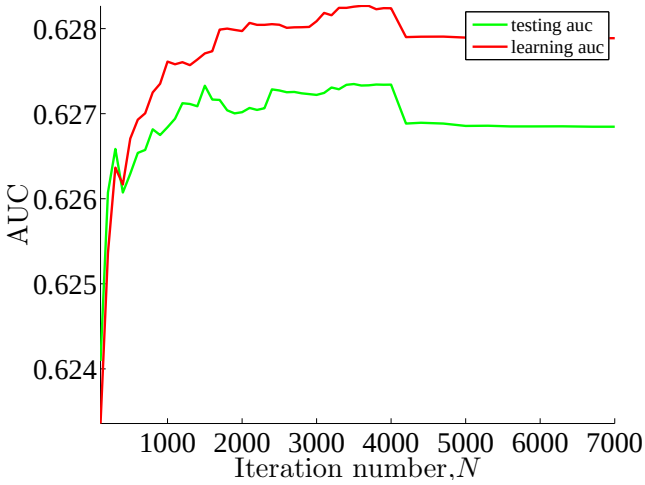
Для данных по потребительским кредитам при $P_E = 0.25$ и $P_R = 0.8$ было отобрано 9 признаков из 24, при $P_E = 0.65$ и $P_R = 0.8$ – 16 признаков. Хотя формально качество классификации после отбора возросло несильно (AUC возросло лишь на 1%), в действительности после отбора алгоритм требует не только меньше входных данных, но и меньше вычислительного времени, что демонстрирует следующая серия графиков. На них показана работа алгоритма на протяжении 5000 итераций из двух начальных приближений к оптимальному вектору весов. В случае с отобранными признаками (при $P_E = 0.65$, $P_R = 0.8$) сходимость заметно более быстрая и монотонная.

На рис. 5 и рис. 6 первым приводится график для исходных признаков, затем для отобранных с помощью шаговой регрессии при $P_E = 0.65$, $P_R = 0.8$.

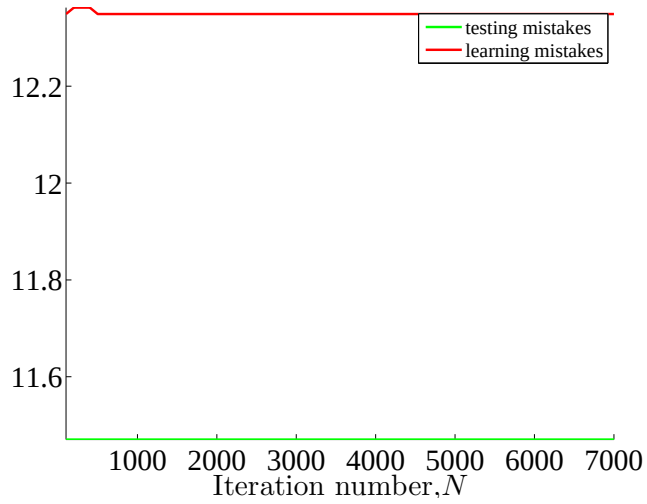
Рис. 5 и рис. 6 демонстрируют, что после 5000 итераций градиентного спуска в случае исходных признаков полученные $\mathbf{w}$ из разных начальных приближений ещё значительно отличаются, а также, что сходимость немонотонна. Напротив, в случае отобранных



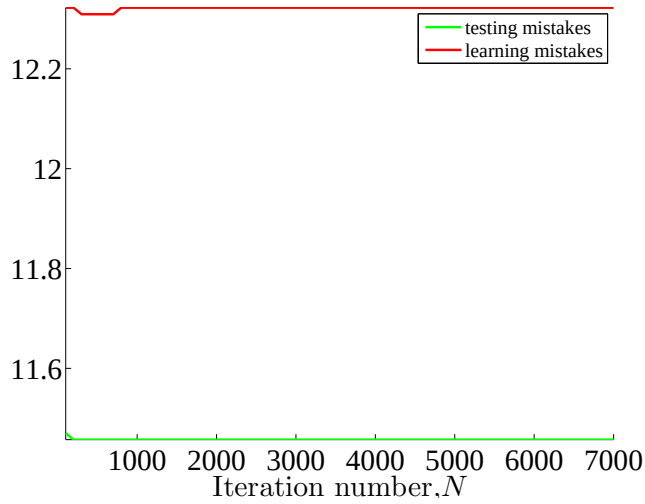
(a) Исходные признаки, $P_E = 0.5$, $P_R = 0.8$



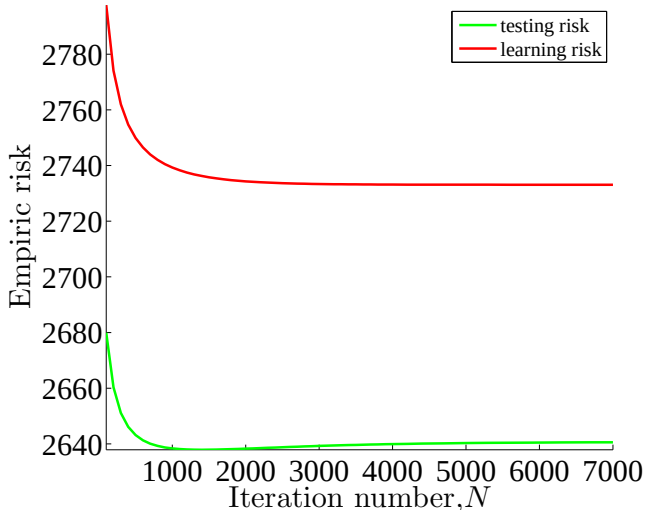
(b) Разделенные признаки, $P_E = 0.5$, $P_R = 0.8$



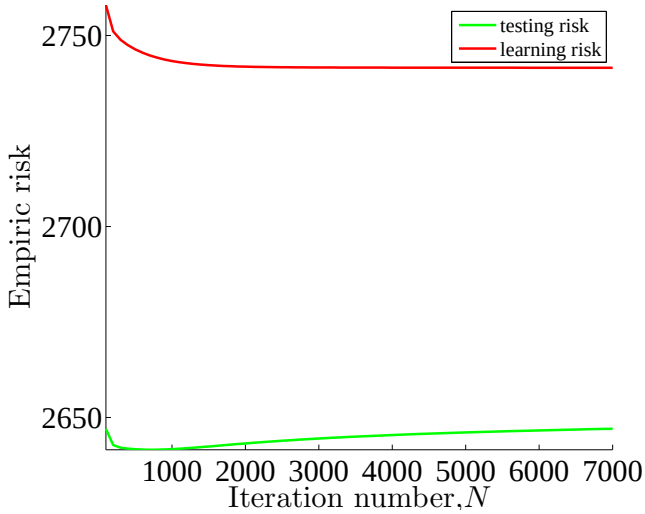
(c) Исходные признаки, $P_E = 0.5$, $P_R = 0.8$



(d) Разделенные признаки, $P_E = 0.5$, $P_R = 0.8$



(e) Исходные признаки, $P_E = 0.5$, $P_R = 0.8$



(f) Разделенные признаки, $P_E = 0.5$, $P_R = 0.8$

Рис. 7. Зависимость площади под *ROC*—кривой, процента ошибок и эмпирического риска от числа итераций градиентного спуска для $P_E = 0.5$, $P_R = 0.8$

признаков сходимость монотонная и после 5000 итераций заметных отличий траекторий эволюции $\mathbf{w}$ не наблюдается.

Маркетинговая кампания

Применим алгоритм отбора признаков к данным об отклике клиентов на маркетинговую кампанию банка [7]. Сравним результат работы алгоритма отбора на исходных признаках и признаках, полученных после работы алгоритма порождения признаков.

Априори можно предположить, что на этих данных качество классификации, выраженное через площадь под ROC-кривой будет ниже, чем для данных о потребительских кредитах, поскольку то, примет ли человек участие в промо-акции банка зависит во многом не от его дохода, места работы и пр., а от того, пожелает ли он того в конкретный момент времени. Однако полученные результаты говорят о практической применимости алгоритма.

Приведем результаты его работы для исходных и разделенных признаков(рис. 7). В последнем случае число отобранных признаков мало отличается от отобранных по исходным, однако качество классификации несколько выше.

Переобучение в обоих случаях наступает примерно после 1000 итераций, когда эмпирический риск $R(\mathbf{w}, T, \mathcal{A})$ для тестовой выборки T начинает расти при продолжающемся снижении эмпирического риска $R(\mathbf{w}, S, \mathcal{A})$ для обучающей выборки.

Заключение

В данной работе рассматривалась задача порождения признаков и выбора оптимального их набора, а также определения весов признаков с целью оценки качества заемщиков. Результаты вычислительного эксперимента показали, что после порождения признаков по описанному в работе алгоритму заметно возрастает по сравнению с исходными признаками скорость сходимости, а также заметно слабее растет норма вектора весов $\mathbf{w}$.

Также была исследована зависимость величины эмпирического риска от параметров шаговой регрессии при отборе признаков. Оказалось, что в рассмотренных примерах рекомендованные в [2] значения параметров не являются оптимальными. Отбор признаков еще более ускоряет сходимость, а она приобретает более монотонный характер.

Литература

[1] N. Siddiqi. *Credit risk scorecards: developing and implementing intelligent credit scoring.* Wiley, 2006.

[2] D.W. Hosmer and Stanley Lemeshow. *Applied logistic regression.* A Wiley-Interscience Publication, 2000.

[3] C.M. Bishop. *Pattern Recognition and Machine Learning.* Springer, 2006.

[4] C.M. Bishop, N.M. Nasrabadi. Pattern recognition and machine learning. *J. Electronic Imaging*, 16(4):049901, 2007.

[5] К.В. Воронцов. *Линейные методы классификации.* MachineLearning.Ru, февраль 2010.

[6] T. Hastie, R. Tibshirani, J.H. Friedman. *The Elements of Statistical Learning.* Springer, 2001.

[7] Данные об отклике клиентов otp-банка на маркетинговую кампанию. http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german/, 2011.

[8] Данные о немецких потребительских кредитах. http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german/, 2000.

[9] А.Г. Сухарев, А.Г. Тимохов, В.В. Федоров. *Курс методов оптимизации.* Физматлит, 2005.

Использование метода главных компонент при построении интегральных индикаторов*

М. М. Медведникова
medvmasha@rambler.ru
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В данной работе рассматривается использование метода главных компонент при построении интегральных индикаторов. Полученные результаты сравниваются с результатами, даваемыми методом расслоения Парето. Строится интегральный индикатор для российских вузов. Для этого используются биографии 30 богатейших бизнесменов России по версии журнала «Forbes» за 2011 год.

Ключевые слова:*интегральный индикатор, экспертные оценки, веса параметров, метод главных компонент, метод расслоения Парето*

Введение

Современная востребованность рейтингов высших учебных заведений обусловлена существованием большого числа вузов. В частности, в России на данный момент их насчитывается более тысячи. Существует достаточно много рейтингов, построенных с использованием различных критериев [1, 2]. В данной работе предлагается в качестве критерия для оценки вуза использовать успешность карьеры выпускников. Предполагается, что чем выше качество образования, тем выше человек продвигается по карьерной лестнице, не зависимо от того, работает он по полученной специальности, или нет.

Для построения рейтингов используются интегральные индикаторы. Построение интегрального индикатора — введение отношения порядка на множестве сравнимых объектов. Предполагается, что каждый объект описан вектором, компоненты которого являются результатами измерений соответствующих показателей. Множество рассматриваемых объектов называется *выборкой*. Выборка полностью описывается матрицей, строками которой являются векторы, сопоставляемые объектам. Все измерения выполнены в линейных шкалах. *Линейная шкала* — это шкала, на которой равным отрезкам соответствует равные абсолютные приращения показателя. *Интегральный индикатор* — скаляр, поставленный в соответствие объекту. Интегральный индикатор для набора объектов — вектор, компоненты которого поставлены в соответствие сравниваемым объектам.

Распространенным алгоритмом [4, 5, 6, 7, 8, 9] построения интегральных индикаторов для объектов, описанных в линейных шкалах, является линейная комбинация значений показателей. Основная задача заключается в определении весов показателей.

Существуют две основные разновидности рассматриваемой задачи. Первая — построение интегрального индикатора методом «с учителем». В этом случае имеются экспертные оценки качества объектов и важности показателей, необходимо согласовать значения интегрального индикатора и весов показателей. Для этого разработаны различные алгоритмы: использующие экспертные оценки качества объектов [3], использующие оценки качества обектов и весов признаков и уточняющие эти оценки [4, 5]. Вторая разновидность — построение индикатора методом «без учителя». Веса вычисляются исходя из некоторого заданного критерия информативности описаний. В этом случае используется метод расслоения Парето [6], вычисления расстояний [7], метод главных компонент [8, 9, 10, 11].

Научный руководитель В. В. Стрижов

В настоящей работе для построения интегрального индикатора будут использованы метод главных компонент и метод расслоения Парето. Метод главных компонент заключается в том, что к множеству описаний обектов применяется преобразование вращения, которое соответствует критерию наибольшей информативности С. Р. Рао [10]. Согласно этому критерию, наибольшая информативность есть минимальное значение суммы квадратов расстояния от описания обектов до их проекций на первую главную компоненту. Приводится подробное изложение теоретического обоснования метода главных компонент в методических целях. Метод расслоения Парето состоит в разделении выборки на слои несравнимых объектов. В статье приведены описание алгоритма и его теоретическое обоснование, базирующееся на [10], для метода главных компонент и описание алгоритма для метода расслоения Парето, также представлены результаты вычислительных экспериментов для рассматриваемых методов и проведено их сопоставление.

Постановка задачи в общем виде

Дана матрица «объекты-признаки» A. Каждая строка $\mathbf{a}_i^T$, $i = 1, ..., p$ этой матрицы — это вектор, описывающий обект. В данной работе предполагается, что в матрице A данные представлены полностью, без пропусков.

Требуется найти отображение

$$F : A \to \mathbf{q},$$

сопостовляющее каждой строке $\mathbf{a}_i^T$ матрицы A интегральный индикатор q_i.

Предложенное ниже обоснование метода главных компонент, используемого при решении данной задачи, является авторской версией изложения [10].

Базис Грама — Шмидта. Пусть $\mathbf{u}^T = (u_1, ..., u_p)$ — p-мерная случайная величина, с нулевым математическим ожиданием и ковариационной матрицей Σ ранга $m \leqslant p$.

$$M(\mathbf{u}) = \mathbf{0}.$$

$$\operatorname{rang}(\Sigma) = m \leqslant p.$$

Введем обозначение:

$$\mathfrak{M}(\mathbf{u}) = \mathfrak{M}(u_1, ..., u_p) = \{c_1 u_1 + ... + c_p u_p \mid c_i \in \mathbb{R}\}.$$

По определению $\mathfrak{M}(\mathbf{u})$ является линейным пространством. Определим скалярное произведение двух элементов y_1, y_2 из $\mathfrak{M}(\mathbf{u})$ следующим образом:

$$\langle y_1, y_2 \rangle = \operatorname{cov}(y_1, y_2) = \mathrm{M}(y_1 y_2).$$

Тогда нормой элемента y_1 является квадратный корень из его дисперсии:

$$\|y_1\| = \sqrt{\mathrm{D}(y_1)} = \sqrt{\mathrm{M}(y_1^2)}.$$

Из линейной алгебры известно, что $\mathfrak{M}(\mathbf{u})$ имеет ортонормированный базис $g_1, ..., g_m$, где $g_1, ..., g_m$ — попарно не коррелированные случайные величины с единичной дисперсией. Тогда каждый элемент пространства $\mathfrak{M}(\mathbf{u})$ может быть представлен в виде:

$$u_i = a_{i1} g_1 + ... + a_{im} g_m, \quad i = 1, ..., p.$$

Или в матричной форме:

$$\mathbf{u} = A\mathbf{g},$$

где $\mathbf{g}^T = (g_1, ..., g_m)$ и матрица $A = \|a_{ij}\|$. При этом элементы ковариационой матрицы Σ выражаются следующим образом:

$$\Sigma_{ij} = \langle u_i, u_j \rangle = \mathrm{cov}(u_i, u_j) = \sum_{k=1}^{m} a_{ik} a_{jk},$$

$$\Sigma = AA^T.$$

Обратно, если $\mathfrak{M}(\mathbf{u}) = \mathfrak{M}(\mathbf{g})$, то вектор $\mathbf{g}$ может быть выражен через вектор $\mathbf{u}$:

$$\mathbf{g} = B\mathbf{u},$$

$$I = B\Sigma B^T,$$

где I — единичная матрица.

Также из линейной алгебры известно, что размерность пространства $\mathfrak{M}(\mathbf{u})$ равна рангу матрицы скалярных произведений элементов $u_1, ..., u_p$, которой в данном случае является матрица Σ. Следовательно, число случайных величин в ортонормированном базисе равно $m = \mathrm{rang}(\Sigma)$. Ортонормированный базис не единственен. Однако некоторые специальные базисы представляют статистический интерес, их мы и будем рассматривать.

Линейный предиктор как проекция. Пусть $P(u_i)$ — проекция u_i на $\mathfrak{M}(u_1, ..., u_{i-1})$. По определению это линейная функция переменных $u_1, ..., u_{i-1}$, которая определяется как

$$P(u_i) = b_1^* u_1 + ... + b_{i-1}^* u_{i-1} = \underset{b_1,...,b_{i-1}}{\arg\min} \|u_i - \sum_{r=1}^{i-1} b_r u_r\|^2 = \underset{b_1,...,b_{i-1}}{\arg\min} \mathrm{M}(u_i - \sum_{r=1}^{i-1} b_r u_r)^2.$$

Следовательно, $P(u_i)$ является линейным предиктором случайной величины u_i, основанным на величинах $u_1, ..., u_{i-1}$, с минимальной среднеквадратичной ошибкой. Более того,

$$u_i - P(u_i) \perp \mathfrak{M}(u_1, ..., u_{i-1}).$$

Поэтому коэффициенты $b_1^*, ..., b_{i-1}^*$ наилучшей линейной функции определяются из условия равенства нулю скалярных произведений разности $u_i - \sum_{r=1}^{i-1} b_r u_r$ и некоторых $u_s, s = 1, ...i-1$. Поскольку скалярное произведение определено как ковариация, то условие имеет вид:

$$\mathrm{cov}(u_s, u_i - \sum_{r=1}^{i-1} b_r u_r) = 0, \quad s = 1, ..., i-1,$$

откуда следует, что

$$\mathrm{cov}[u_s, u_i - P(u_i)] = 0, \quad s = 1, ..., i-1,$$

что равносильно следующему:

$$\mathrm{cov}(u_s, u_i) = \mathrm{cov}[u_s, P(u_i)], \quad s = 1, ..., i-1.$$

В таком случае имеем:

$$\mathrm{cov}[u_i - P(u_i), u_j - P(u_j)] = \mathrm{cov}(u_i, u_j) - \mathrm{cov}[u_i, P(u_j)] + \mathrm{cov}[P(u_i), P(u_j)] - \mathrm{cov}[P(u_i), u_j] = 0, \quad i \neq j.$$

Если обозначить остаток $u_i - P(u_i)$ через $u_{i,12\ldots i-1}$, то предыдущая формула эквивалентна утверждению, что остатки

$$u_1,\ u_{2,1},\ u_{3,12},\ \ldots\ , u_{p,12\ldots p-1}$$

попарно не коррелированы.

Ортонормированный базис Грама — Шмидта. Пусть $t_{ii} = \|u_i - P(u_i)\|$ — норма (квадратный корень из дисперсии) остатка $u_{i,12\ldots i-1}$. Рассмотрим

$t_{11}g_1 = u_1 = u_1,$
$t_{22}g_2 = u_2 - P(u_2) = u_2 - t_{21}g_1,$
.....
$t_{pp}g_p = u_p - P(u_p) = u_p - t_{p1}g_1 - \ldots - t_{p,p-1}g_{p-1}.$

Отметим, что проекция случайной величины u_i на $\mathfrak{M}(u_1, \ldots, u_{i-1})$ может быть выражена через $g_1, \ldots, g_{i-1}$, а коэффициенты последовательно определены с помощью процесса ортогонализации Грама — Шмидта. Если $t_{ii} = 0$, то u_i полностью определяется предыдущими случайными величинами. В противном случае, если $t_{ii} \neq 0$, то

$$g_i = \frac{1}{t_{ii}}[u_i - P(u_i)] = \frac{1}{t_{ii}}u_{i,12\ldots i-1},$$

так что $\|g_i\| = 1$. Определенные таким образом величины g_i, соответствующие ненулевым t_{ii}, составляют ортонормированный базис. Обратные соотношения имеют вид:

$u_1 = t_{11}g_1,$
$u_2 = t_{21}g_1 + t_{22}g_2,$
.....
$u_p = t_{p1}g_1 + t_{p2}g_2 + \ldots + t_{pp}g_p$

и могут быть переписаны в виде:

$$\mathbf{u} = T\mathbf{g},$$

где T — нижняя треугольная матрица. Тогда можно выразить ковариационную матрицу Σ через матрицу T следующим образом:

$$\Sigma = TT^T.$$

Анализ главных компонент

Рассмотрим p случайных величин $\mathbf{u} = (u_1, \ldots, u_p)$ с ковариационной матрицей Σ. Пусть $\lambda_1 \geqslant \ldots \geqslant \lambda_p$ — собственные числа, а $\mathbf{p}_1, \ldots, \mathbf{p}_p$ — соответствующие им собственные векторы матрицы Σ. Тогда, как известно из линейной алгебры,

$$\mathbf{p}_i^T\Sigma\mathbf{p}_i = \lambda_i; \quad \mathbf{p}_i^T\Sigma\mathbf{p}_j = 0, \quad i \neq j.$$

Рассмотрим случайные величины, получающиеся в результате преобразования

$$y_i = \mathbf{p}_i^T\mathbf{u}, \quad i = 1, \ldots, p.$$

Обозначим через $\mathbf{y}$ вектор новых случайных величин и через P ортогональную матрицу со столбцами из собственных векторов матрицы Σ:

$$\mathbf{y}^T = (y_1, \ldots, y_p), \quad P = (\mathbf{p}_1, \ldots, \mathbf{p}_p).$$

Тогда вектор $\mathbf{y}$ можно получить из вектора $\mathbf{u}$ с помощью ортогонального преобразования:

$$\mathbf{y} = P\mathbf{u}.$$

Случайная величина y_i называется i-той главной компонентой случайной величины $\mathbf{u}$.

Свойства главных компонент

1. *Главные компоненты не коррелированы. Дисперсия i-той главной компоненты равна* λ_i.
   Это следует из соотношений:

$$\mathrm{D}(\mathbf{p}_i^T\mathbf{u}) = \langle \mathbf{p}_i^T\mathbf{u}, \mathbf{p}_i^T\mathbf{u}\rangle = \langle p_{i1}u_1 + ... + p_{ip}u_p,\ p_{i1}u_1 + ... + p_{ip}u_p\rangle = \mathbf{p}_i^T\Sigma\mathbf{p}_i = \lambda_i;$$

$$\mathrm{cov}(\mathbf{p}_i^T\mathbf{u}, \mathbf{p}_j^T\mathbf{u}) = \mathbf{p}_i^T\Sigma\mathbf{p}_j = 0, \quad i \neq j.$$

   Таким образом, линейное преобразование $\mathbf{y} = P\mathbf{u}$ переводит коррелированное множество случайных величин в некоррлерованное.
2. *Пусть* $g_i = \frac{1}{\sqrt{\lambda_i}}\boldsymbol{p}_i^T\boldsymbol{u}$ *для* $\lambda_i \neq 0$ *и* $rang(\Sigma) = r$, *так что отличными от нуля оказываются первые* r *собственных чисел матрицы* Σ. *Тогда* $g_1, ..., g_r$ — *ортонормированный базис случайной величины* $\boldsymbol{u}$.
3. *Пусть* $\boldsymbol{b}$ — *произвольный вектор, такой что* $\|\boldsymbol{b}\| = 1$. *Тогда дисперсия* $D(\boldsymbol{b}^T\boldsymbol{u})$ *достигает максимума при* $\boldsymbol{b} = \boldsymbol{p}_1$ *и этот максимум равен* λ_1*:*

$$\boldsymbol{p}_1 = \arg\max_{\|\boldsymbol{b}\|=1} D(\boldsymbol{b}^T\boldsymbol{u});$$

$$\lambda_1 = \max_{\|\boldsymbol{b}\|=1} D(\boldsymbol{b}^T\boldsymbol{u}).$$

4. Следствия из пунктов 1-3:
   —
$$\min_{\|\mathbf{b}\|=1} \mathrm{D}(\mathbf{b}^T\mathbf{u}) = \lambda_p = \mathrm{D}(\mathbf{p}_p^T\mathbf{u})$$
   —
$$\max_{\|\mathbf{b}\|=1, \mathbf{b}\perp\mathbf{p}_1,...,\mathbf{p}_{l-1}} \mathrm{D}(\mathbf{b}^T\mathbf{u}) = \lambda_l = \mathrm{D}(\mathbf{p}_l^T\mathbf{u})$$
   — Пусть $\mathbf{b}_1, ..., \mathbf{b}_k$ — множество ортогональных вектров с единичной нормой. Тогда

$$\lambda_1 + ... + \lambda_k = \max_{\mathbf{b}_1,...,\mathbf{b}_k} [\mathrm{D}(\mathbf{b}_1^T\mathbf{u}) + ... + \mathrm{D}(\mathbf{b}_k^T\mathbf{u})] = \mathrm{D}(\mathbf{p}_1^T\mathbf{u}) + ... + \mathrm{D}(\mathbf{p}_k^T\mathbf{u}).$$

5. **Утверждение**
   Пусть $\boldsymbol{b}_1^T\boldsymbol{u}, ..., \boldsymbol{b}_k^T\boldsymbol{u}$ — k *линейных функций случайной величины* $\boldsymbol{u}$ *и* σ_i^2 — *остаточная дисперсия в предсказании* u_i *с помощью наилучшего линейного предиктора, основанного на* $\boldsymbol{b}_1^T\boldsymbol{u}, ..., \boldsymbol{b}_k^T\boldsymbol{u}$. *Тогда*

$$\min_{\boldsymbol{b}_1,...,\boldsymbol{b}_k} \sum_{i=1}^{p} \sigma_i^2$$

   достигаетсяв случае, если множество $\boldsymbol{b}_1^T\boldsymbol{u}, ..., \boldsymbol{b}_k^T\boldsymbol{u}$ *эквивалентно множеству* $\boldsymbol{p}_1^T\boldsymbol{u}, ..., \boldsymbol{p}_k^T\boldsymbol{u}$, *то есть каждая из величин* $\boldsymbol{b}_i^T\boldsymbol{u}$ *есть линейная комбинация первых* k *главных компонент.*

   Доказательство

По определению

$$\sigma_i^2 = \|u_i - \sum_{j=1}^{k} \langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle \mathbf{b}_j^T \mathbf{u}\|^2 = \mathrm{D}(\langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle \mathbf{b}_j^T \mathbf{u}) = \mathrm{D}(u_i - P(u_i)).$$

Без ограничения общности можно считать $\mathbf{b}_1^T \mathbf{u}, ..., \mathbf{b}_k^T \mathbf{u}$ некоррелированными функциями с единичной дисперсией. Для оптимального решения эти функции должны быть линейно независимы.
Проведем преобразование выражения для σ_i^2.

$$\sigma_i^2 = \mathrm{D}(u_i - P(u_i)) = \mathrm{D}(u_i) + \mathrm{D}[P(u_i)] - 2\mathrm{cov}[u_i, P(u_i)] =$$

$$= \mathrm{D}(u_i) + \mathrm{D}[\sum_{j=1}^{k} \langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle \mathbf{b}_j^T \mathbf{u}] - 2 \sum_{j=1}^{k} \langle u_i, \langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle \mathbf{b}_j^T \mathbf{u} \rangle$$

Т.к. все $\mathbf{b}_i^T \mathbf{u}$ были заменены на некоррелированные величины с единичной дисперсией, то можно продолжить следующим образом:

$$\sigma_i^2 = \mathrm{D}(u_i) + \sum_{j=1}^{k} \langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle^2 - 2 \sum_{j=1}^{k} \langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle^2 =$$

$$= \mathrm{D}(u_i) - \sum_{j=1}^{k} \langle \mathbf{b}_j^T \mathbf{u}, u_i \rangle^2$$

Обозначим $\mathrm{D}(u_i)$ как σ_{ii}, а i-тый столбец матрицы Σ как Σ_i. Тогда получаем, что

$$\sigma_i^2 = \sigma_{ii} - \sum_{j=1}^{k} [\mathrm{cov}(u_i, \mathbf{b}_j^T \mathbf{u})]^2 =$$

$$= \sigma_{ii} - \sum_{j=1}^{k} [b_{j1} \langle u_i, u_1 \rangle + ... + b_{jp} \langle u_i, u_p \rangle]^2 =$$

$$= \sigma_{ii} - \sum_{j=1}^{k} [\mathbf{b}_j^T \Sigma_i]^2 = \sigma_{ii} - \sum_{j=1}^{k} [\mathbf{b}_j^T \Sigma_i \Sigma_i^T \mathbf{b}_j].$$

Теперь просуммируем все остаточные дисперсии.

$$\sum_{i=1}^{p} \sigma_i^2 = \sum_{i=1}^{p} \sigma_{ii} - \mathbf{b}_1^T (\sum_{i=1}^{p} \Sigma_i \Sigma_i^T) \mathbf{b}_1 - ... - \mathbf{b}_k^T (\sum_{i=1}^{p} \Sigma_i \Sigma_i^T) \mathbf{b}_k =$$

$$= \mathrm{Tr}(\Sigma) - \mathbf{b}_1^T (\Sigma \Sigma^T) \mathbf{b}_1 - ... - \mathbf{b}_k^T (\Sigma \Sigma^T) \mathbf{b}_k.$$

Для того чтобы минимизировать $\sum \sigma_i^2$ нужно найти максимальное значение суммы

$$\mathbf{b}_1^T (\Sigma \Sigma^T) \mathbf{b}_1 + ... + \mathbf{b}_k^T (\Sigma \Sigma^T) \mathbf{b}_k$$

при условиях

$$\mathbf{b}_i^T \Sigma \mathbf{b}_i = 1; \quad \mathbf{b}_i^T \Sigma \mathbf{b}_j = 0, \quad i \neq j,$$

которые обеспечивают, что случайные величины $\mathbf{b}_1^T\mathbf{u}, ..., \mathbf{b}_k^T\mathbf{u}$ не коррелированы и имеют единичную дисперсию. В таком случае при оптимальном выборе векторов $\mathbf{b}_i$ они являются собственными векторами для характеристического уравнения

$$\det(\Sigma\Sigma - \lambda\Sigma) = 0.$$

Но собственные числа и векторы такого уравнения совпадают с собственными числами и векторами уравнения

$$\det(\Sigma - \lambda I) = 0.$$

■

Интерпретация главных компонент

Как показано в [10], полученный результат дает возможность интерпретировать главные компоненты следующим образом. Предположим, что мы хотим заменить p-мерную случайную величину на $k < p$ линейных функций, теряя не слишком много информации. Эффективность выбора этих функций зависит от того, в какой степени они дают возможность реконструировать p первоначальных случайных величин. Один из методов реконструкции случайной величины u_i состоит построении ее наилучшего линейного предиктора на основе k линейных функций. В этом случае эффективность предиктора может быть измерена с помощью остаточной дисперсии σ_i^2. Полная мера эффективности предиктора равна $\sum \sigma_i^2$. *Наилучшим выбором линейных функций, для которых $\sum \sigma_i^2$ минимальна, является выбор первых k главных компонент случайной величины $\boldsymbol{u}$.*

Уточнение постановки задачи

Введем следующие обозначения:

$\hat{\Sigma} = A^TA$ — оценка ковариационной матрицы объектов;

$\mathbf{y}$ — вектор главных компонент;

W — весовая матрица (матрица преобразования вращения). Она является ортогональной, то есть $I = W^TW$;

$\mathbf{q}$ — интегральный индикатор.

Обозначим $Z = WA$ матрицу, состоящую из столбцов $(\mathbf{z}_1, ..., \mathbf{z}_p)$. Для нахождения первой главной компоненты необходимо найти такие линейные комбинации строк матрицы A, что векторы-столбцы матрицы Z обладали бы наибольшей дисперсией.

Требуется найти:

$$W_* = \arg\min_{W^TW=I} \sum_{i=1}^{p} \|\mathbf{a}_i - (\mathbf{a}_i, \mathbf{w})\mathbf{w}\|^2,$$

где $\mathbf{a}_i$ — векторы-строки матрицы «объекты-признаки» A, $\mathbf{w}$ — один из столбцов матрицы W, причем $\|\mathbf{w}\| = 1$.

Такая постановка задачи в силу доказанного С. Р. Рао [10] утверждения эквивалентна следующей:

$$W_* = \arg\max_{W^TW=I} \sum_{j=1}^{p} D\mathbf{z}_j.$$

При этом интегральный индикатор строится в виде:

$$\mathbf{q} = A\mathbf{w},$$

где $\mathbf{w}$ —один из столбцов матрицы W_*.

Описание алгоритма

Как было показано, для нахождения первой главной компоненты, используемой при построении интегрального индикатора, в качестве матрицы преобразования вращения нужно рассматривать матрицу, составленную из собственных векторов ковариационной матрицы вектора признаков:

$$W_* = P = \{\mathbf{p}_1, \mathbf{p}_2, ...\mathbf{p}_n\},$$

где $\mathbf{p}_1$, $\mathbf{p}_2$, ... $\mathbf{p}_n$ — собственные векторы ковариационной матрицы $\hat{\Sigma}$.

Направление первой главной компоненты определяет собственный вектор, соответсвующий максимальному собственному числу. Если $\lambda_1, \lambda_2, ...\lambda_n$ — собственные числа матрицы $\hat{\Sigma}$ и для них выполнено:

$$\lambda_1 \geqslant \lambda_2 \geqslant ... \geqslant \lambda_n,$$

то искомый вектор $\mathbf{w}$ определяется как

$$\mathbf{w} = \mathbf{p}_1.$$

Метод расслоения Парето

В предположении, что признаки могут быть измерены в ранговых шкалах, используем для построения интегрального индикатора метод расслоения Парето.

Уточнение постановки задачи. Введем отношение доминирования на наборе объектов $\{\mathbf{a}_i\}_{i=1}^m$. Объект $\mathbf{a}_i$ доминирует объект $\mathbf{a}_k$ ($\mathbf{a}_i \succ \mathbf{a}_k$, $\mathbf{a}_i \neq \mathbf{a}_k$), если все компоненты вектора $\mathbf{a}_i$ больше или равны соответсвующим компонентам вектора $\mathbf{a}_k$:

$$a_{ij} \geqslant a_{kj},\ j = 1, ...n.$$

Определим Парето-оптимальный фронт как набор недоминируемых объектов.

Требуется разделить имеющийся набор объектов на Парето-слои из недоминируемых объектов.

Описание алгоритма. Рассмотрим строки матрицы «объекты-признаки» как набор сравниваемых объектов. Будем отсекать Парето-слои, начиная с нижнего. Обозначим P_l l-тый Парето-слой.

$$P_l = \{\mathbf{a} \in A \mid \neg\exists \mathbf{x} \in A : x_i \leqslant a_i, i = 1, ...n\}$$

Исключим из матрицы полученный слой перейдем к получению следующего. Процесс остановится, когда матрица A станет пустой.

Алгоритм Парето-расслоения описан в [12].

Вычислительный эксперимент

Метод главных компонент на модельных данных

В ходе эксперимента использовались синтетические данные (рис. 1), которые подбирались вручную так, чтобы точки, соответствующие описаниям объектов, находились на плоскости вблизи одной прямой.

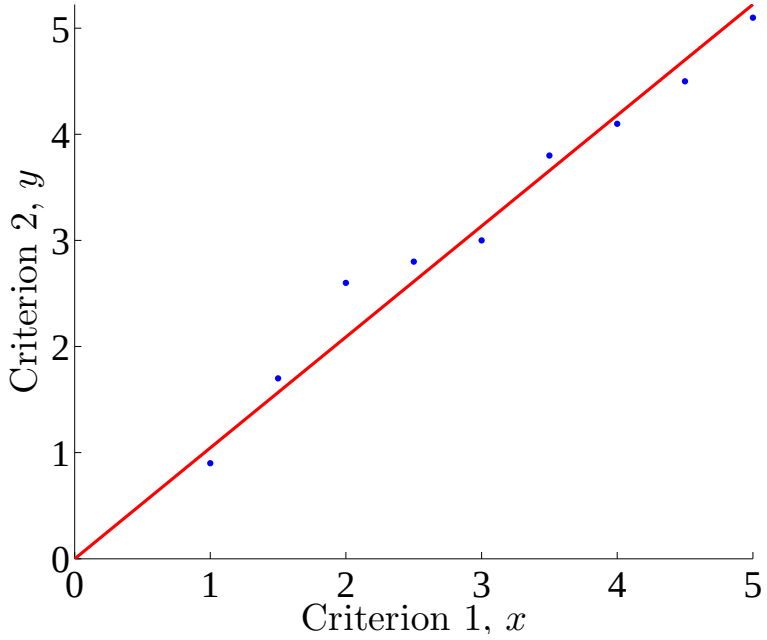


Рис. 1. Построение первой главной компоненты для синтетических данных. Синие точки соответствуют описаниям объектов, красная прямая показывает направление главной компоненты.

Метод главных компонент на реальных данных

В настоящей работе для построения интегрального индикатора для вузов предлагается разбивать биографии выпускников на группы, сооветвующие следующим сферам деятельности:

1. Образование, наука, инновационные разработки;
2. Бизнес, экономика;
3. Политика, государственное управление, деятельность в общественных организациях;
4. Культура и искусство;
5. Спортивная карьера.

В каждой группе будет построен интегральный индикатор методом главных компонент. Интегральный индикатор вуза будет определяться как среднее арифметическое из индикаторов его выпускников.

Эксперимент проводился для выборки в сфере деятельности «Бизнес, экономика». В выборку вошли описания биографий 30 богатейших бизнесменов России по версии журнала «Forbes» за 2011 год. Для визуализации результатов использовались попарно 3 признака, оказавшиеся наиболее близкими к главной компоненте. На рисунке 2 представлены в виде точек входные данные, красные линии — направления первых главных компонент для пары признаков.

Так же при помощи вычисления коэффициента ранговой корреляции Спирмена между полученным интегральным индикатором и доходом бизнесменов была проверена гипотеза о том, что определяющую роль в индикаторе играет доход. Гипотеза не была подтверждена: значение коэффициента Спирмена равно 0.4794, что соответствует слабой прямой связи между величинами.

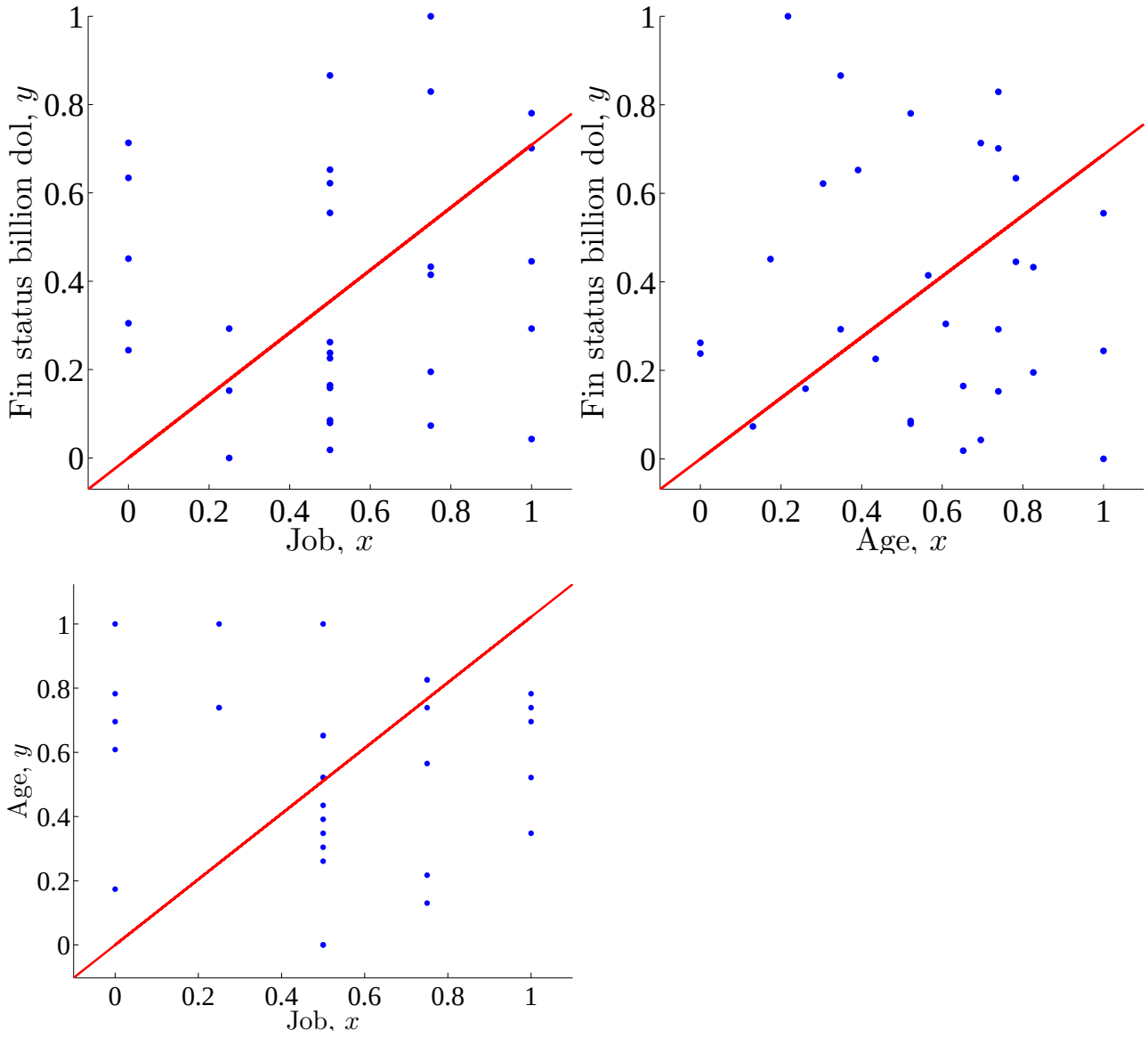


Рис. 2. Построение первой главной компоненты. Синие точки соответствуют описаниям объектов, красная прямая показывает направление главной компоненты. Оси нормированы.

Сравнение результатов для разных способов построения интегральных индикаторов

Для возможности визуализации интегральные индикаторы строились для трех признаков. Результаты, полученные с помощью метода главных компонент, представлены в том же формате, что и в предыдущем случае (рис. 3). Результаты, полученные при помощи расслоения Парето, представлены на рисунке 4, где различными цветами обозначены разные слои.

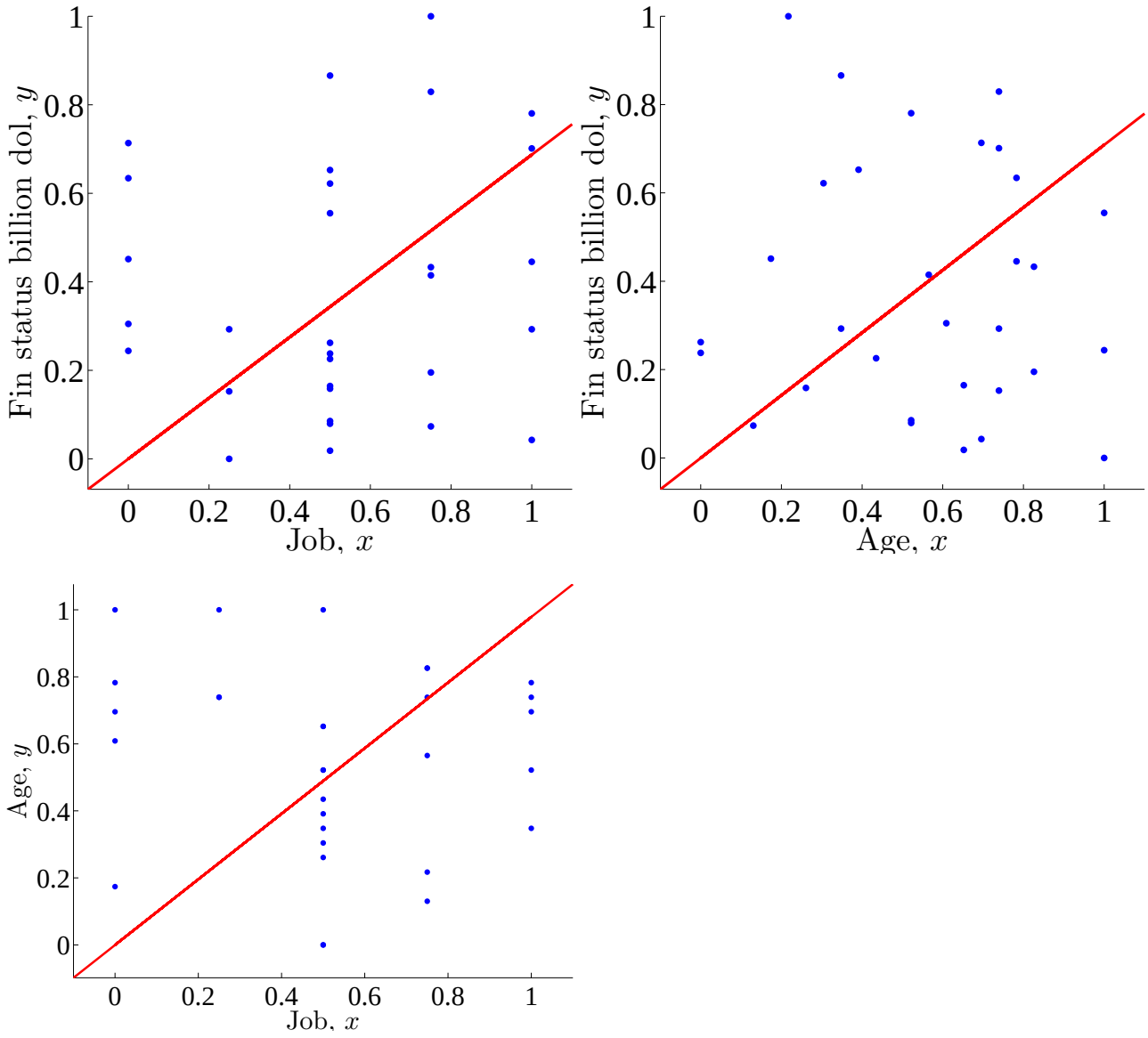


Рис. 3. Построение первой главной компоненты. Синие точки соответствуют описаниям объектов, красная прямая показывает направление главной компоненты. Оси нормированы.

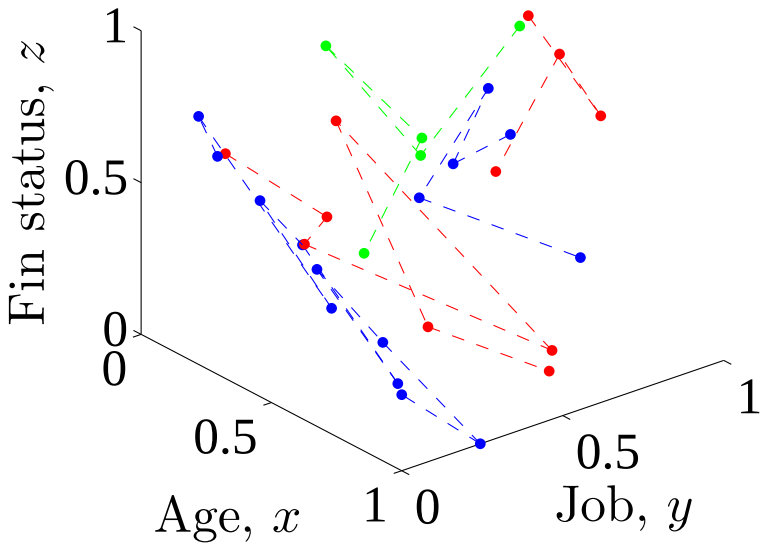


Рис. 4. Парето-слои. Оси нормированы.

Сравнение полученных результатов представлено на рисунке 5, где по горизонтальной оси отложены интегральные индикаторы объектов, полученные методом расслоения Парето, а по вертикальной оси — индикаторы, полученные методом главных компонент. Как можно видеть, результаты, даваемые разными методами, отличаются, но между ними прослеживается линейная зависимость.

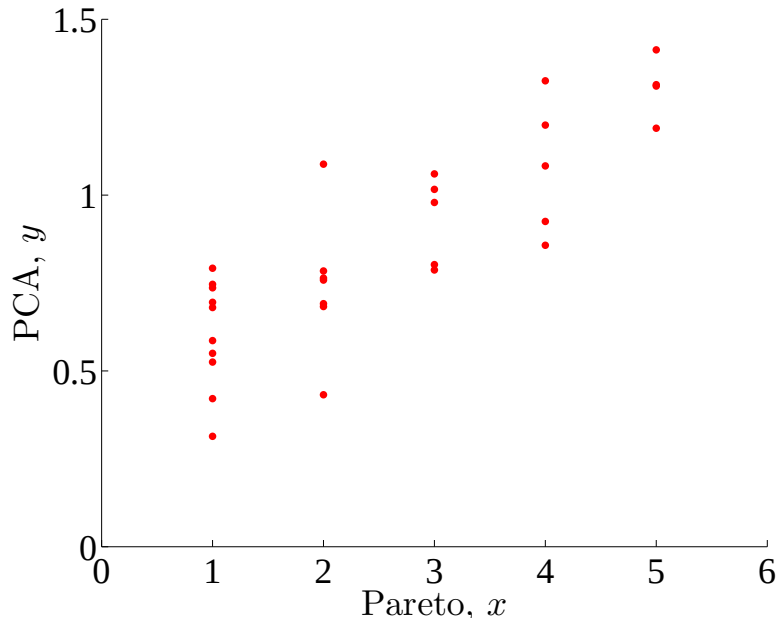


Рис. 5. Сравнение полученных результатов.

Заключение

В работе были приведены описание и теоретическое обоснование метода главных компонент, а также описание алгоритма расслоения Парето для построения интегральных индикаторов. Проведен вычислительный эксперимент и сравнение полученных результатов. Сделан вывод, что результаты, даваемые рассмотренными двумя методами связаны между собой линейно.

Литература

[1] О. М. Карпенко, М. Д. Бершадская. *Международный рейтинг университетов Webometrics: основные идеи, индикаторы, результаты*, Педагогические Измерения, 2010, №2.

[2] С.Ċ. Донецкая. *Российский подход к ранжированию ведущих университетов мира*, ЭКО, 2009, №9: 137-150

[3] С. А. Айвазян, В. М. Бухштабер, И. С. Енюков, Л. Д. Мешалкин. *Прикладная статистика. Классификация и снижение размерностей*, Финансы и статистика, 1989, с.334, 421-424.

[4] М. П. Кузнецов, В. В. Стрижов. *Уточнение ранговых экспертных оценок с использованием монотонной интерполяции*. Всероссийская конференция «Математические методы распознавания образов», сборник докладов. МАКС-Пресс, 2011.

[5] В. В. Стрижов. *Уточнение экспертных оценок с помощью измеряемых данных*, Заводская лаборатория. Диагностика материалов, 2006, с.59-64.

[6] Strijov, V. *Expert estimations concordance for biosystems under extreme conditions. Notes on applied mathematics*, Moscow, Coumpiting Center of RAS, 2002.

[7] Vadim Strijov and Goran Granic and Jeljko Juric and Branka Jelavic and Sandra Antecevic Maricic. *Integral indicator of ecological impact of the Croatian thermal power plants*, Energy, 2011, №7: 4144-4149.

[8] Strijov, V. and Shakin, V. *Index construction: the expert-statistical method*, Proc. Conference on Sustainability Indicators and Intelligent Decisions, 2003, 56-57.

[9] В. В. Стрижов, Т. В. Казакова. *Устойчивые интегральные индикаторы с выбором опорного множества описаний*, Заводская лаборатория. Диагностика материалов. 2007, 72-74.

[10] С. Р. Рао. *Линейные и статистические методы и их применения*, Наука, 1968, 530-533.

[11] I. T. Jolliffe *Principal Conmponent Analysis*, Springer, 2002.

[12] М. М. Медведникова. *Алгоритм Парето-расслоения*,
https://mlalgorithms.svn.sourceforge.net/svnroot/mlalgorithms/Medvednikova2012PCA/code/Pareto, 2012.

Кластеризация коллекции текстов*

А. А. Романенко
angriff07@gmail.com
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В работе предлагается метод кластеризации текстовой коллекции с помощью стандартных метрических алгоритмов, например, K-means. Для этого вводится функция расстояния между текстами, учитывающая «схожесть» лексики используемой в тексте. В работе также исследуется соответствие между введенным расстоянием на множестве реальных текстов и близостью тематик этих текстов. Возможность кластеризации и соответствие ее результатов с заранее известным распределением текстов по тематике исследована в вычислительном эксперименте на синтетической коллекции текстов.

Ключевые слова: *информационный поиск, метрические алгоритмы кластеризации, кластеризация текстов, K-means.*

Введение

Кластеризация текстов может применяться для выделения из текстовой коллекции групп текстов одинаковой тематики. Эта задача относится к задачам поиска скрытой неструктурированной информации. Из-за больших размеров текстовых коллекций и из-за субъективности восприятия читателя темы текста оценить качество кластеризации сложно. Поэтому пока нет общепринятого функционала качества кластеризации текстовых коллекций и алгоритма, являющегося абсолютно лучшим.

Кластеризацию текстов можно провести с помощью вероятностных методов [1], например с помощью вероятностного латентного семантического анализа (англ. *PLSA — probabilistic latent semantic analysis*) [2] или латентного размещения Дирихле (англ. *LDA — latent Dirichlet allocation*) [3]. В данной работе ставиться задача кластеризации текстовой коллекции с помощью стандартных метрических алгоритмов кластеризации, например K-means [4], FOREL [5], C-means [6]. Для этого на множестве документов предлагается ввести функцию расстояния.

Пусть есть некоторое множество слов русского языка, каждое из которых хотя бы раз встретилось в одном из документов текстовой коллекции. Назовем это множество словарем. В данной работе под документом будем понимать неупорядоченное множество слов из словаря. Слова в документе могут повторяться. Тогда каждому документу можно поставить в соответствие вектор, содержащий информацию о словарном составе документа. Размерность этого вектора равна количеству слов в словаре. Тогда расстояние между документами можно ввести как расстояние между векторами, соответствующими этим документам.

Для улучшения работы алгоритма предлагается сделать предобработку текстов. Во-первых, предлагается привести все слова к своей начальной лексической форме и удалить все знаки препинания. Во-вторых, предлагается убрать из текста слова, встречающиеся в нем малое количество раз, а также слова, встречающиеся в большинстве текстов (стоп-слова). В-третьих, предлагается воспользоваться методикой TFIDF (от англ. *TF — term frequency, IDF — inverse document frequency*), описанной в [7].

Научный руководитель В. В. Стрижов

Для тестирования предложенного метода кластеризации было проведены эксперименты на синтетических данных и на реальных текстах. Цель эксперимента на синтетических данных — проверить возможность кластеризации и то, насколько эта кластеризация соответствовала заранее известному распределения текстов по тематике. Словарь, используемый для генерации текстов, был разбит на множества слов, относящихся к определенной теме. Текст относился к той теме, к которой относилось большинство слов текста.

Цель эксперимента на реальных данных — изучить, отражают ли метрики, рассматриваемые в работе, действительные расстояния между текстами, т. е. сравнить расстояния между реальными текстами на введенных метриках с экспертными расстояниями между ними. Тексты, используемые в эксперименте, — это работы студентов Восточной экономико-юридической гуманитарной академии.

Далее будет представлена математическая постановка задачи и предлагаемое решение. Затем будет представлены результаты вычислительного эксперимента с использованием различных функций расстояний на множестве документов на синтетической коллекции документов.

Постановка задачи и предлагаемое решение задачи

Пусть $W = \{w_1, \ldots, w_{|W|}\}$ — заданное множество слов, словарь. Документом d назовем множество слов из W, порядок которых не важен:

$$d = \{w_j\}, \text{ где } w_j \in W \ \text{ — } j\text{-ое слово в документе } d, \ j = 1, \cdots, |d|.$$

Таким образом, документ имеет модель «мешка слов» [8].

Пусть $D = \{d_1, \ldots, d_{|D|}\}$ — множество всех текстовых документов, k — заданное число кластеров, на которое требуется разбить множество D.

Требуется задать функцию расстояния на множестве документов:

$$\rho(d_i, d_j) : D \times D \longrightarrow \mathbb{R}_+,$$

и провести кластеризацию текстовой коллекции.

Предлагается удалить из текстов стоп-слова и слова, встречающиеся не более одного раза в тексте, как шумовую составляющую. Стоп-слово формально определим как слово из некоторого заранее заданного списка S.

Представим каждый преобразованный документ в виде вектора:

$$\mathbf{d}_i = \begin{pmatrix} n(d_i, w_1) \\ \vdots \\ n(d_i, w_j) \\ \vdots \\ n(d_i, w_{|W|}) \end{pmatrix}, \tag{1}$$

где $n(d_i, w_j)$ — число вхождений слова $w_j \in W$ в текст d_i.

Далее используя это представление документа, ввести расстояние между документами как расстояние между векторами:

$$\rho(d_i, d_j) = \rho(\mathbf{d}_i, \mathbf{d}_j).$$

В качестве функции расстояния $\rho(\mathbf{x}, \mathbf{y})$ между векторами $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ можно взять метрику Минковского для различных значений p, в частности расстояние городских кварталов

при $p = 1$

$$\rho_1(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} |x_k - y_k|,$$

Евклидово расстояние при $p = 2$

$$\rho_2(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2}$$

или расстояние Чебышева при $p = \infty$

$$\rho_\infty(\mathbf{x}, \mathbf{y}) = \max_{k=1,\ldots,n} |x_k - y_k|.$$

Предлагается, используя функцию расстояния $\rho(\mathbf{d}_i, \mathbf{d}_j)$, провести кластеризацию текстовой коллекции одним из метрических алгоритмов кластеризации (K-means, C-means, и т.п.).

Каждый метод кластеризации можно рассматривать как точный или приближённый алгоритм поиска оптимума некоторого функционала. Если y_j — номер кластера, к которому отнесет j-ый документ алгоритм кластеризации, то можно ввести следующие функционалы качества:

$$F_0 = \frac{\sum_{i<j} [y_i = y_j] \rho(\mathbf{d}_i, \mathbf{d}_j)}{\sum_{i<j} [y_i = y_j]} \to \min,$$

т.е. нужно минимизировать среднее внутрикластерное расстояние,

$$F_1 = \frac{\sum_{i<j} [y_i \neq y_j] \rho(\mathbf{d}_i, \mathbf{d}_j)}{\sum_{i<j} [y_i \neq y_j]} \to \max,$$

т.е. нужно максимизировать среднее межкластерное расстояние.

Чтобы учесть и внутрикластерное и межкластерное расстояние можно ввести функционал F :

$$F = \frac{F_0}{F_1} \to \min .$$

Вычислительный эксперимент на синтетических данных

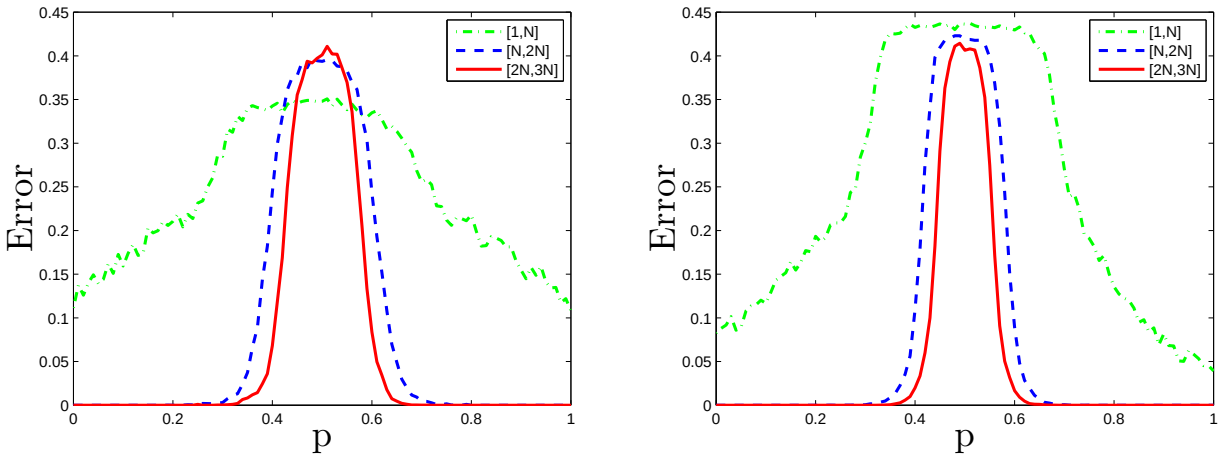
Формирование текстовой коллекции. Для иллюстрации работы алгоритма проведен эксперимент кластеризации на два кластера на синтетической коллекции документов. В качестве словаря был взят список из 50 слов. Предполагалось, что каждое слово из словаря относилось либо к теме 1, либо к теме 2. Каждый документ порождался следующим образом:

1. За длину документа принималось некоторое целое произвольное число из отрезка $[A, B]$.
2. Каждое слово документа, относящегося к теме 1, с вероятностью p выбиралось произвольным образом из списка слов, относящихся к теме 1, и с вероятностью $(1 - p)$ — из списка слов, относящихся к теме 2.
3. Каждое слово документа, относящегося к теме 2, с вероятностью p выбиралось произвольным образом из списка слов, относящихся к теме 2, и с вероятностью $(1 - p)$ — из списка слов, относящихся к теме 1.

Таким образом была получена текстовая коллекция состоящая из 100 текстов: 50 текстов, относящихся к теме 1, и 50 текстов, относящихся к теме 2.

Результаты кластеризации. Сопоставив каждому документу из текстовой коллекции вектор описанным выше способом (1), используем алгоритм кластеризации K-means для кластеризации на два кластера. Ниже представлены зависимости ошибки кластеризации $Error$ от параметра p для метрик ρ_1 и ρ_2 при разных A и B, влияющих на длину документов. Под ошибкой кластеризации здесь понимается доля документов, ошибочно попавших в кластер с документами другой тематики.

Стоит отметить, что так как в алгоритме используется рандомизация, то графики зависимостей получаются не гладкими. Чтобы этого избежать, проводилось 500 экспериментов, а затем ошибка кластеризации усреднялась.



(a) Расстояние городских кварталов

(b) Евклидово расстояние

Рис. 1. Зависимость ошибки кластеризации от параметра p, если длина текста — произвольное число из отрезка $[1, N]$, $[N, 2N]$ или $[2N, 3N]$.

Из рис. 1 видно, что наибольшая ошибка кластеризации наблюдается при p близком к 0.5, что соответствует наибольшему количеству шума в документе. Если же $p \approx 1$ или $p \approx 0$, то ошибка кластеризации мала. Как видно на рис. 1, для метрики ρ_2 с увеличением длины документов при фиксированном p ошибка кластеризации уменьшается. Это же справедливо и для метрики ρ_1, если p не лежит в окрестности 0.5. Если же $p \approx 0.5$, то чем больше размер документа, тем больше ошибка кластеризации. При этом видно, что при использовании метрики ρ_2 ошибка кластеризации меньше, чем при использовании метрики ρ_1.

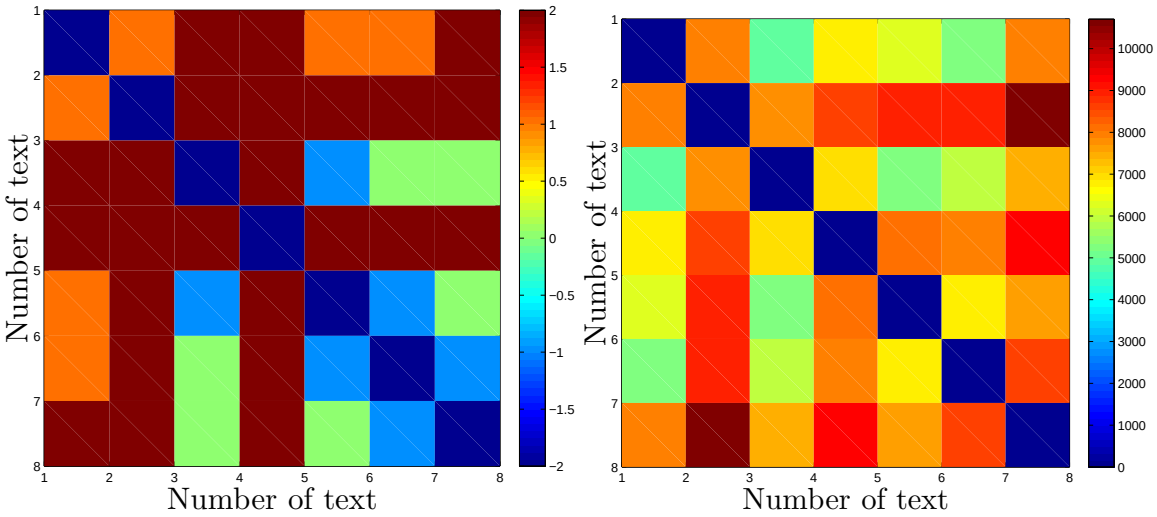
Код для проведения эксперимента можно взять здесь [9].

Вычислительный эксперимент на реальных данных

Описание эксперимента. Для того чтобы убедиться, что рассматриваемые метрики адекватно описывают расстояние между реальными документами, был проведен следующий эксперимент. Из коллекции работ студентов Восточной Экономико-юридической Гуманитарной Академии было взято восемь произвольных текстов. Оказалась, что тексты имели следующие темы:

1. «Система защиты трудового права»

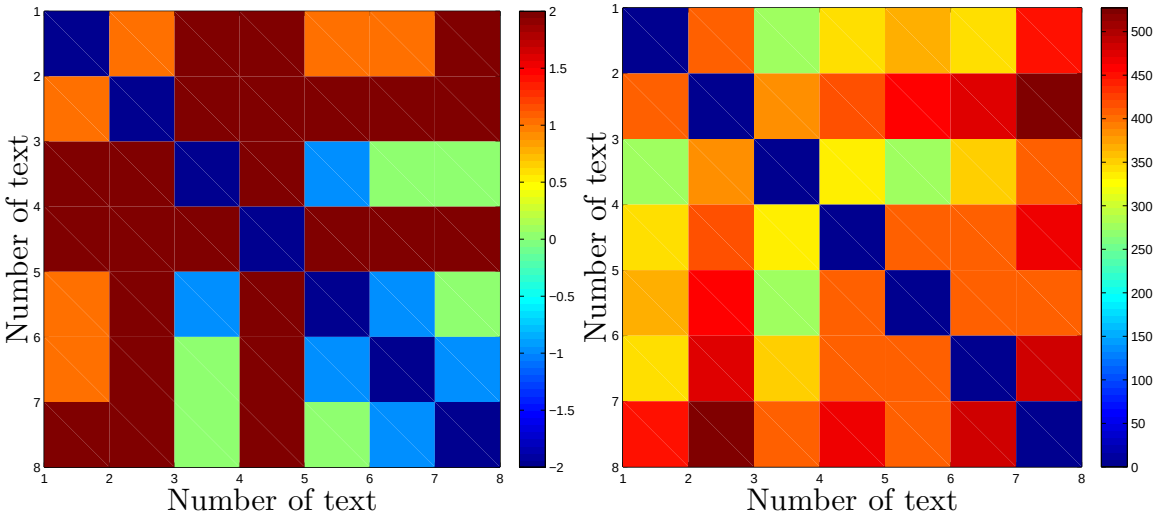
2. «Происхождение государства и права»
3. «Предпринимательство. Сущность, формы и современные особенности»
4. «Оценка состояния новорожденного по его поведению и мимике»
5. «Экономика организации предприятия»
6. «Государственное и муниципальное правление объектами здравоохранения»
7. «Анализ качества продукции ООО «Оренбургский хлебозавод»»
8. «Правоотношения»



(a) Экспертная матрица расстояний

(b) Матрица расстояний между текстами с метрикой ρ_1

Рис. 2. Матрицы расстояний между текстами



(a) Экспертная матрица расстояний

(b) Матрица расстояний между текстами с метрикой ρ_2

Рис. 3. Матрицы расстояний между текстами

Данные восемь текстов были предложены эксперту для определения степени отличия тематики содержимого текстов. Эксперт давал оценку в следующей лингвистической шкале:

-2 — «содержимое текстов очень похоже по тематике»
-1 — «содержимое текстов скорее похоже по тематике, чем отлично»
0 — «трудно определить, похоже содержимое или отлично»
1 — «содержимое текстов скорее отлично по тематике, чем похоже»
2 — «содержимое текстов сильно отличается по тематике»

Также эти тексты были представлены как элементы векторного пространства описанным выше способом (1), и было посчитано расстояние между ними на основании метрик ρ_1 и ρ_2.

Результаты эксперимента и вывод. На рис. 2 и 3 изображены для для сравнения экспертная матрица расстояний и матрицы расстояний между документами, подсчитанных с метриками ρ_1 и ρ_2.

Из рисунков 2, 3 видно, что резкие различия в тематике рассматриваемые метрики выявить могут. Так, например, по мнению эксперта, текст №2 сильно отличается от всех текстов. Действительно, он находится на большом расстоянии от оставшихся документов. Но с определением тонких различий в тематике они справляются хуже.

Код для проведения эксперимента и данные можно взять здесь [9].

Заключение

В работе описан способ представления документа как элемента векторного пространства. Это дает возможность ввести функцию расстояния между документами и кластеризовать коллекцию документов на основе метрических алгоритмов кластеризации. В эксперименте на синтетической коллекции документов исследуется соответствие результатов кластеризации коллекции с заранее известным распределением текстов по тематике.

Также в работе исследуется соответствие между введенным расстоянием на множестве реальных документов и близостью тематик реальных документов. Для этого проведен эксперимент с реальными текстами.

Литература

[1] A. Daud, J. Li, L. Zhou, F. Muhammad. *Knowledge discovery through directed probabilistic topic models: a survey.* Frontiers of Computer Science in China. 2010.

[2] Hofmann T. *Probabilistic latent semantic indexing.* SIGIR '99. New York, NY, USA: ACM, 1999.

[3] Blei D. M., Ng A. Y., Jordan M. I. *Latent dirichlet allocation.* 2003.

[4] Hartigan J. A., Wong M. A. *Algorithm as 136: A k-means clustering algorithm.* 1978.

[5] Н.Г.Загоруйко, В.Н.Ёлкина, Г.С.Лбов. *Алгоритмы обнаружения эмпирических закономерностей.* Новосибирск: Наука, 1985.

[6] Pal N. R., Bezdek J. C. *On cluster validity for the fuzzy c-means model.* 1995.

[7] Manning C. D., Raghavan P., Schütze H. *Introduction to Information Retrieval.* Cambridge: Cambridge University Press, 2008.

[8] Lewis D. D. *Naive (bayes) at forty: The independence assumption in information retrieval.* Springer Verlag, 1998.

[9] А.А.Романенко *Кластеризация коллекции текстов: вычислительный эксперимент.* 2012. `http://bit.ly/IT2OXW`.

Локальные методы прогнозирования с выбором преобразования*

С. В. Цыганова
schiavoni@mail.com
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В работе описан алгоритм локального прогнозирования с учетом преобразований, позволяющий выявить похожие во введеной метрике интервалы временного ряда. Рассмотренно понятие инвариантных преобразований, их обнаружение и выбор наиболее подходящих для решения задачи прогнозирования. Работа алгоритма проиллюстрирована на данных потребления электроэнергии и на синтетических данных.

Ключевые слова: *локальное прогнозирование, функция расстояния, временной ряд, инвариантное преобразование*

Задачи прогнозирования временных рядов имеют множество приложений в различных областях, таких как экономика, физика, медицина. Их решением является прогноз на недалекое будущее по уже известным значениям прогнозируемого ряда в предыдущие моменты времени. Данная работа посвящена методу локального прогнозирования временных рядов. Для построения прогноза используются только те части временного ряда, которые близки к конечному отрезку всего временного ряда. Близкими считаются те отрезки временных рядов, функция близости для которых мала. Для определения близких отрезков в работе исследуется линейное преобразование (сжатие, сдвиг), инварианты преобразований и функция «близости» отрезков временных рядов, которая будет являться одним из критериев адекватной работы построенного алгоритма прогнозирования. Общий локальный метод прогнозирования основан на идеях, описанных в работе Дж. Макнеймса [1] и Ю.И. Журавлева [2].

Для нахождения близких интервалов использован метод «ближайшего соседа», успешно применяемый к широкому классу прикладных задач, таких как прогнозирование объемов продаж, прогнозирование цен на электроэнергию, постановка диагноза по биоритмам человека.

Сложности при построении алгоритма – это учет пропусков в предоставленных данных. В данной работе считается, что данные представлены без пробелов.

Проверка алгоритма будет производиться при помощи скользящего контроля, т.е. прогноз будет сравниваться с реальными значениями.

Вся работа разделена на четыре главы. Первая глава – это математическая постановка задачи. Во второй главе описывается алгоритм преобразования и прогнозирования с некоторыми математическими выкладками. Третья глава – вычислительный эксперимент для двух временных рядов (синтетического и потребления энергии [6]) и исследование эффективности алгоритма. В последней главе сформулирован общий вывод.

Научный руководитель В. В. Стрижов

Постановка задачи

Будем рассматривать одномерные временные ряды — ряды, в которых каждому моменту времени сопоставляется вещественное число.

$$\{t_1, t_2, \ldots, t_n\} \to \{x_1, x_2, \ldots, x_n\}$$

Требуется предсказать следующие l значений последовательности $\{x_{n+1}, x_{n+2}, \ldots, x_{n+l}\}$, которые будут определяться значением предыстроии $\{x_{n-L+1}, x_{n-L+2}, \ldots, x_n\}$ длины L. Для этого необходимо выполнить следующий алгоритм:

1. Выделить во всем временном ряде вектора длины

$$r : r_{min}, \ldots, r_{max},$$

которые после линейных преобразований A (сжатие,сдвиг) похожи на предысторию $S = \{x_{n-L+1}, x_{n-L+2}, \ldots, x_n\}$.

2. Найти и исследовать инварианты преобразований между двумя близкими векторами временного ряда и с их помощью найти те самые «похожие» вектора.

3. Критерий близости играет функция близости двух векторов **a**, **b** — в данной работе это взвешенная метрика Евклида:

$$D_{WE}(a, b) = \sqrt{(a-b)^T \Lambda^2 (a-b)}.$$

4. Задача формулируется следующим образом:

$$\text{dist}(A(x_{k-r+1}, x_{k-r+2}, \ldots, x_k), (x_{n-L+1}, x_{n-L+2}, \ldots, x_n)) \to min,$$

где $A(x_{k-r+1}, x_{k-r+2}, \ldots, x_k)$ – преобразованный близкий вектор, а $(x_{n-L+1}, x_{n-L+2}, \ldots, x_n)$ – вектор предыстории.

5. Для отыскания k близких векторов используем метод k ближайших соседей. Пусть

$$\{A_1(x_{i_1-r+1}, \ldots, x_{i_1}), \ldots, A_k(x_{i_k-r+1}, \ldots, x_{i_k})\} - -$$

это k ближайших соседей для предыстории $\{x_{n-L+1}, x_{n-L+2}, \ldots, x_n\}$.

Прогноз вычисляется как среднее k векторов:

$$\{A_1(x_{i_1+1}, \ldots, x_{i_1+l}), \ldots, A_k(x_{i_k+1}, \ldots, x_{i_k+l})\},$$

где среднее вычисляется как взвешенное среднее арифметическое:

$$(x_{i_1+1}, \ldots, x_{i_1+l}) = \frac{\sum_{j=1}^{k} w_j A_j(x_{i_j+1}, \ldots, x_{i_j+l})}{\sum_{j=1}^{k} w_j},$$

$$w_j = \left(1 - \frac{d_{ij}^2}{d_{i_{k+1}}^2}\right)^2,$$

где $d_{i_{k+1}}^2$ – расстояние до $k+1$ ближайшего соседа.

Описание алгоритма

Алгоритм включает в себя следующие этапы:

1. Нахождение преобразования φ по оси OX и выбор потенциальных соседей – векторов $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m$. Для этого во всем временном ряде X находятся точки экстремумов. Далее находятся такие векторы, чтобы точки экстремальных значений этих векторов до точек экстремальных значений предыстории имели минимальное расстояние в выбранной метрике. Максимальное отклонение, допускаемое алгоритмом – заданный параметр ε. Из этого условия для каждого i-го потенциального соседа находится коэффициент a_{0_i} растяжения по OX, а выбранные векторы $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m$ становятся потенциальными соседями. Последующие значения – $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m$ – потенциальным прогнозом в зависимости от близости соседа. Количество потенциальных соседей прямопропорционально параметру ε: чем меньше параметр, тем меньше потенциальных соседей выделяет алгоритм.

Таким образом мы находим соседние векторы $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m$ для предыстории $\{x_{n-L+1}, x_{n-L+2}, \ldots, x_n\}$ у временных рядов не только с постоянным, но и с изменяющимся периодом.

2. Для каждого потенциального соседа минимизируется функция близости и находятся коэффициенты b_{0_i} растяжения по OY для каждого близкого вектора. Пусть $\mathbf{y}$ – данный вектор временного ряда, а $\mathbf{x}$ – вектор временного ряда, который необходимо преобразовать по оси OY, чтобы получить наиболее близкий к $\mathbf{y}$ вектор. Тогда требуется найти минимум следующей функции:

$$F = \sum_{t=1}^{l} (y_t - (a + bx_t))^2.$$

Необходимые условия экстремума (более подробно смотри [3]):

$$\begin{cases} \frac{dF}{da} = -2\sum_{t=1}^{l}(y_t - a - bx_t) = 0, \\ \frac{dF}{db} = -2\sum_{t=1}^{l} x_t(y_t - a - bx_t) = 0. \end{cases}$$

Раскроем скобки и получим систему уравнений:

$$\begin{cases} al + b\sum_{t=1}^{l} x_t = \sum_{t=1}^{l} y_t, \\ a\sum_{t=1}^{l} x_t + b\sum_{t=1}^{l} x_t^2 = \sum_{t=1}^{l} x_t y_t. \end{cases}$$

Решениями системы являются:

$$\begin{cases} b = \frac{l\sum_{t=1}^{l} x_t y_t - (\sum_{t=1}^{l} x_t)(\sum_{t=1}^{l} y_t)}{n\sum_{t=1}^{l} x_t^2 - (\sum_{t=1}^{l} x_t)^2}, \\ a = \frac{1}{l}\sum_{t=1}^{l} y_t - \frac{1}{l}\sum_{t=1}^{l} x_t b. \end{cases}$$

3. Соседи сортируются по значению функции близости. Далее выбираются k самых близких (k – заданное число). Их потенциальный прогноз усредняется (чем меньше значение функции близости для соседа, тем больший вклад дает k-й потенциальный прогноз в усреднение).

4. Построение прогноза и вычисление ошибки с помощью скользящего контроля. Для сравнения всех прогнозов в работе используется суммарная абсолютная ошибка отклонения прогноза от действительных значений. Обозначим $\mathbf{f} = (f_{n+1}, \ldots, f_{n+l})$ – точное значение временного ряда и $\tilde{\mathbf{f}} = (\tilde{f}_{n+1}, \ldots, \tilde{f}_{n+l})$ – полученный алгоритмом прогноз. Тогда качество алгоритмов сравнивается при помощи следующей величины:

$$E = \frac{1}{l} \sum_{j=1}^{l} \left| f_{n+j} - \tilde{f}_{n+j} \right|.$$

Этот функционал ошибки зависит только от абсолютного отклонения прогноза от точных значений временного ряда и не зависит от их величины.

Вычислительный эксперимент и эффективность

Данный метод прогнозирования предназначен для прогнозирования временных рядов с переменным периодом, что дает алгоритму преимущества перед алгоритмом, использующим преобразования сжатия и сдвига по оси OY и описанным в работе В. Федоровой [4].

Рассмотрим следующий пример – спрогнозируем модельный временной ряд :

$$y = sin(x^2)$$

Сравним работу следующих двух алгоритмов – с использованием преобразования по OX и без:

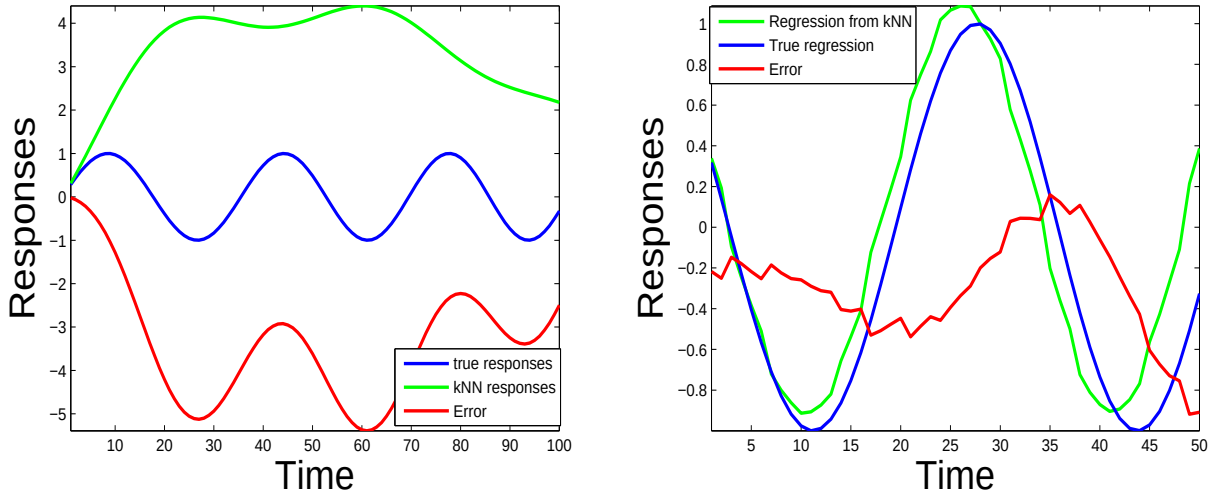


Рис. 1. Прогноз с использованием преобразования по OX (справа) и без (слева)

Теперь сравним работу алгоритмов на реальных данных – будем прогнозировать потребление энергии на один день вперед. График потребления электроэнергии за последние несколько дней (625 временных точек) выглядит следующим образом:

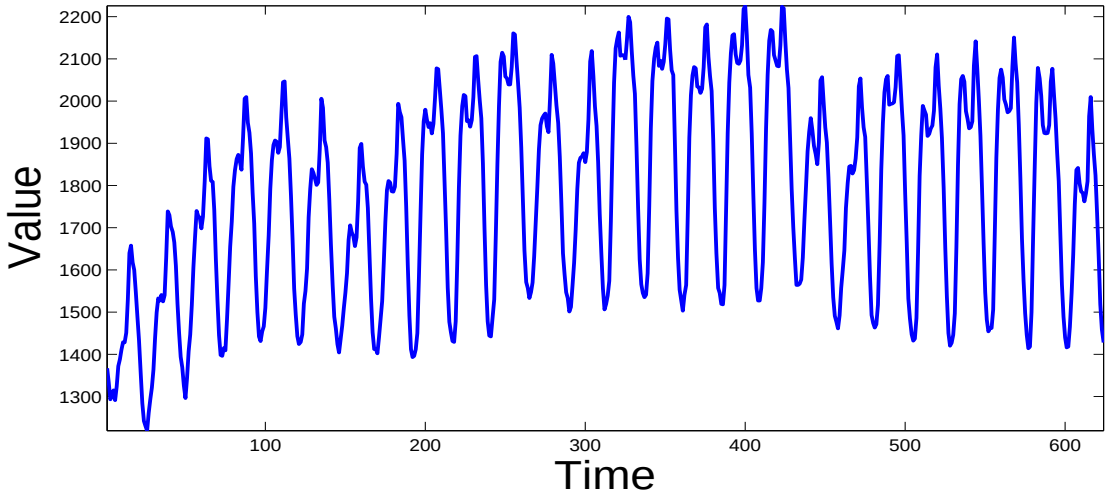


Рис. 2. график потребления электроэнергии за несколько дней

Результат работы алгоритмов:

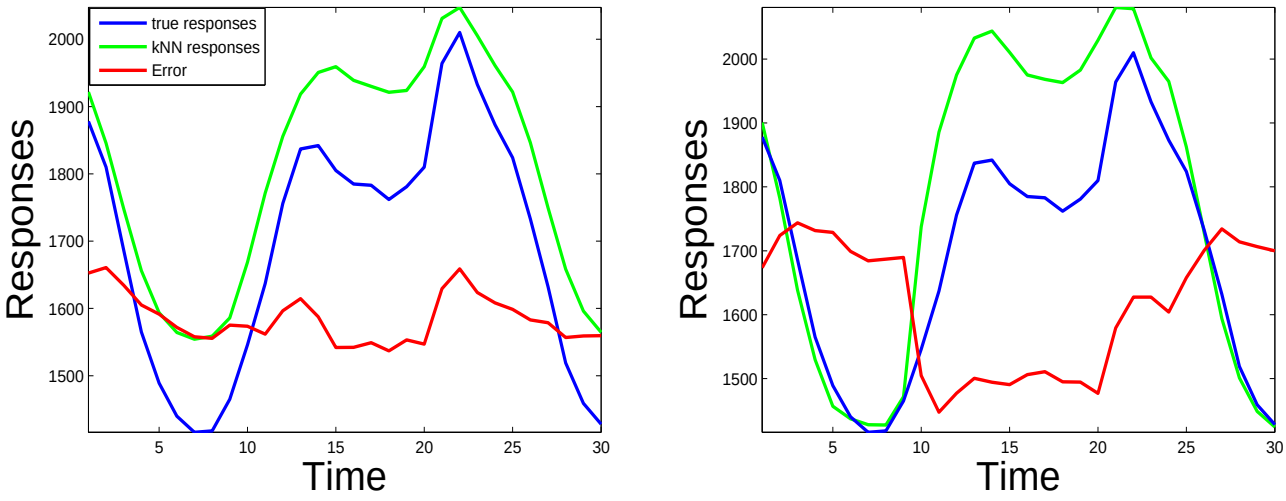


Рис. 3. Прогноз с использованием преобразования по OX (справа) и без (слева)

Видно, что уже на такой небольшой выборке заметно улучшение прогноза по сравнению с алгоритмом, не использующим преобразование по оси OX – в особенности на первых точках прогноза.

Рассмотрим теперь в 10 раз больший временной ряд (тот же самый временной ряд, но с большей историей) и получим следующие результаты:

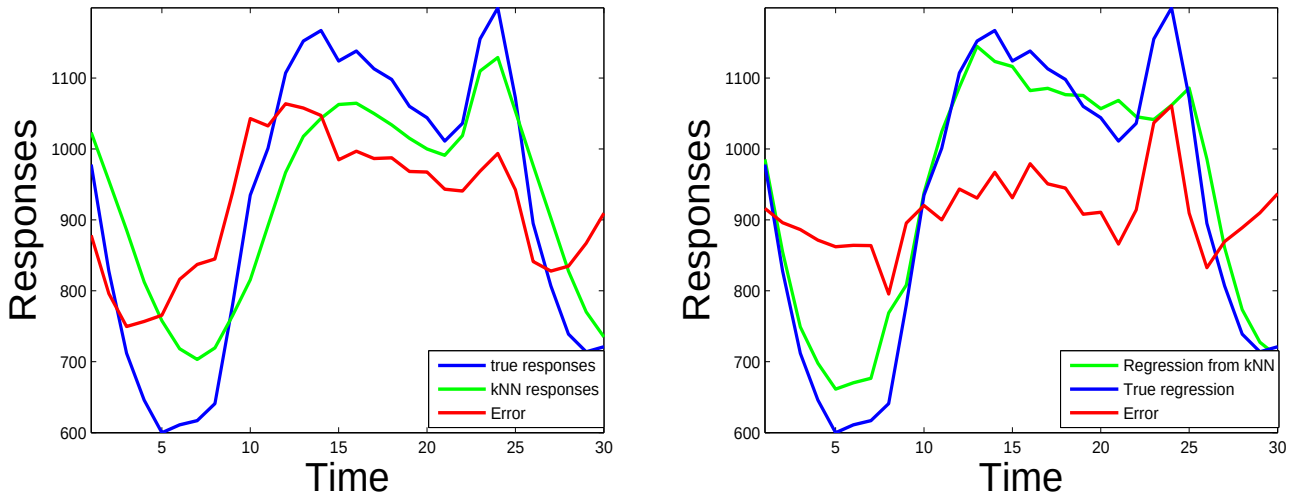


Рис. 4. Прогноз с использованием преобразования по OX (справа) и без (слева)

Алгоритм, использующий преобразование по оси OX прогнозирует намного лучше, нежели алгоритм, не использующий это преобразование. Кроме того, на такой большой выборке первый алгоритм затрачивает значительно меньшее время и ресурсов вычислительной машины, так как ищет коэффициенты преобразований не для всех возможных интервалов временного ряда, а для уже отобранных потенциальных соседей.

Точность прогноза сильно зависит от выбора значений параметров – это количество соседей k, длина предыстории L и параметр ε (максимальная ошибка при совмещении точек экстремумов потенциальных соседей и предыстории). Как уже было сказано во главе «Описание алгоритма», количество потенциальных соседей m пропорционально значению параметра ε. Алгоритм отбирает k наилучших из m потенциальных соседей после преобразований. Кажется, что если увеличивать параметр ε, то это не повлияет на результат – лучшие останутся лучшими и качество прогноза не изменится. Для опровержения этого был проведен следующий эксперимент, в котором исследовался функционал ошибки E прогноза от изменяющегося параметра ε . Результаты эксперимента представлены на следующем графике:

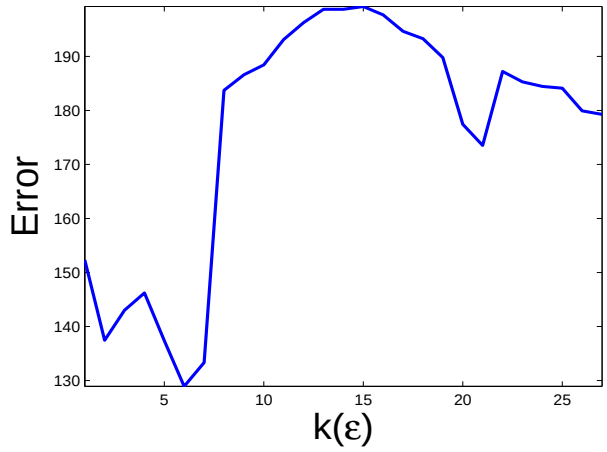


Рис. 5. Значение функционала ошибки от параметра ε

При увеличении параметра ε функционал ошибки E возрастает, что ещё раз подтверждает предположение о том, что преобразование по оси OX имеет большое значение.

Заключение

Итак, в работе был рассмотрен алгоритм локального прогнозирования, использующий афинные преобразования (сжатие, сдвиг) временного ряда по осям OX и OY и основанный на методе kNN (k ближайших соседей). Алгоритм был протестирован на модельных и реальных данных и показал хорошие результаты. В ходе сравнения работы алгоритма с алгоритмом, использующим преобразования только по оси OY, выяснилось, что преобразование по оси OX играет достаточно большую роль в прогнозировании методом kNN, не только улучшая качество прогноза, но и уменьшая затрачиваемые вычислительные ресурсы.

Литература

[1] McNames J.,*Innovations in local modeling for time series prediction* // Ph.D. Thesis, Stanford University, 1999.

[2] Журавлев, Ю. И., Рязанов, В. В., и Сенько, О. В. *Распознавание. Математические методы. Программная система. Практические применения.* // Фазис, Москва, 2005.

[3] Магнус, Я. Р., Катышев, П. К., Пересецкий, А. А. *Эконометрика* // Дело, 2004, стр. 34-37

[4] Федорова, В. П., *Локальные методы прогнозирования временных рядов* // Москва, 2009.

[5] Воронцов, К. В. Курс лекций *Математические методы обучения по прецедентам*

[6] Временные ряды прогнозирования электроэнергии *http://www.neural-forecasting-competition.com*

[7] Дуда, Р., Харт, П. *Распознавание образов и анализ сцен* // Мир, Москва, 1976

Многоуровневая классификация при обнаружении движения цен*

А. А. Кузьмин
senatormipt@gmail.com

В данной работе рассматривается один из возможных методов прогнозирования, основанный на модели логистической регрессии. Предлагается способ разметки пучка временных рядов и построения матрицы объект — признак. Алгоритм проверяется на синтетических пучках временных рядов вида зашумленных синусов и периодических трапеций. Как вариант практического применения, алгоритм тестируется на данных о потреблении электроэнергии.

Ключевые слова: *логистическая регрессия, разметка временных рядов, потребление электроэнергии.*

Введение

Временным рядом называют последовательность, упорядоченную по времени. Это могут быть значения биржевых индексов, цены на определенный товар, характеристики пациента, среднесуточная температура в городе и т.д.[4]. Пучком временных рядов называется набор зависимых временных рядов, например метеорологические данные: температура, давление, скорость и направление ветра [3], [4]. Таким образом, имея все эти данные за некоторый прошедший период, мы гораздо точнее сможем предсказать значение температуры, нежели если бы мы имели лишь данные о температуре. Предполагая зависимость всех временных рядов в исследуемом пучке, будем предсказывать значение одного ряда, основываясь на предыстории всего пучка.

Мы будем рассматривать лишь пучки временных рядов одинаковой длины, представляющие собой дискретную последовательность с одинаковым шагом по времени. В данной работе ставится задача прогнозирования разметки: возрастет или уменьшится его значение в следующий момент. Для этого проводится разметка ряда и строится матрица объект — признак, как, например, в [3],[4], [5]. В качестве модели будем рассматривать логистическую регрессию (см. [1],[6]).

Постановка задачи

Пусть у нас имеется N временных рядов длиной T с шагом по времени t. Для удобства пронормируем шаг, т.е. возьмем $t = 1$ и аналогично пересчитаем T. Будем называть каждый из рядов — $a_i, i \in \{1, 2 \dots N\}$, значение ряда a_i в момент времени t будем обозначать $a_{i,t}$. Таким образом, имеем матрицу с неизвестным столбцом $a_{i,T+1}$, где каждая строка — временной ряд. Пусть, для определенности, нам надо предсказать поведение ряда a_1 в момент $T + 1$.

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,T-1} & a_{1,T} & a_{1,T+1} \\ a_{2,1} & a_{2,2} & \dots & a_{2,T-1} & a_{2,T} & a_{2,T+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{N,1} & a_{N,2} & \dots & a_{N,T-1} & a_{N,T} & a_{N,T+1} \end{pmatrix}$$

Определим множество элементов разметки ряда как: $\mathcal{M} = \{+1, 0, -1\}$. Разметим интересующий нас ряд a_1, причем разметку будем проводить следующим образом: вместо

Научный руководитель В. В. Стрижов

элемента $a_{1,t}$ будем ставить $+1$, если следующий элемент этого ряда $a_{1,t+1} > a_{1,t}$, 0 — если $a_{1,t+1} = a_{1,j}$ и -1, если $a_{1,t+1} < a_{1,j}$. Получим ряд, состоящий из -1, 0 и 1:

$$\mathbf{A} = \begin{pmatrix} 0 & -1 & \dots & 1 & -1 & 0 \\ a_{2,1} & a_{2,2} & \dots & a_{2,T-1} & a_{2,T} & a_{2,T+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{N,1} & a_{N,2} & \dots & a_{N,T-1} & a_{N,T} & a_{N,T+1} \end{pmatrix}.$$

Глубиной лагирования Δ назовем отступ по времени. Задавая ее, мы предполагаем, что на значение в момент времени t интересующего нас ряда влияют в различной степени значения всего пучка на временном отрезке $[t-\Delta, t-1]$. Их мы будем использовать в качестве признаков. Поставим в соответствие каждому значению исследуемого ряда $a_{1,t}$, где $t \in \{\Delta+1, \dots T\}$ матрицу:

$$\mathbf{A}_t = \begin{pmatrix} a_{1,t-\Delta} & a_{1,t-\Delta+1} & \dots & a_{1,t-2} & a_{1,t-1} \\ a_{2,t-\Delta} & a_{2,t-\Delta+1} & \dots & a_{2,t-2} & a_{2,t-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{N,t-\Delta} & a_{N,t-\Delta+1} & \dots & a_{N,t-2} & a_{N,t-1} \end{pmatrix}.$$

Для получения строки признаков x_t векторизуем ее:

$$\mathbf{x}_t = \begin{pmatrix} a_{1,t-\Delta} & a_{2,t-\Delta} & \dots & a_{N,t-\Delta} & a_{1,t-\Delta+1} & \dots & a_{N,t-1} \end{pmatrix}.$$

Значение правильного ответа y_t для этого набора признаков — $a_{1,t}$. Теперь имеем $T-\Delta$ обучающих наборов типа объект — признак или матрицу размером $\Delta * (T-\Delta)$

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_{T-\Delta} \end{pmatrix}^T$$

и столбец ответов для нее $\mathbf{y} = \begin{pmatrix} y_1 & y_2 & \dots & y_{T-\Delta} \end{pmatrix}^T$. Введем также вектор весов как $\mathbf{w} = \begin{pmatrix} w_1 & w_2 & \dots & w_{\Delta * N} \end{pmatrix}$. Для нахождения параметров будем использовать логистическую регрессию, согласно которой

$$P(y|\mathbf{x}) = \sigma(\langle \mathbf{w}, \mathbf{x} \rangle y),$$

где $\sigma(z)$ — сигмоидная функция:

$$\sigma(z) = \frac{1}{1+\exp(-z)}.$$

Значение $z = \langle \mathbf{w}, \mathbf{x}_i \rangle y$ будем называть отступом и обозначать $M_i(\mathbf{w})$. Критерием качества модели является значение логарифма правдоподобия:

$$L(\mathbf{w}, \mathbf{X}) = \sum_{i=1}^{T-\Delta} \log\left(\sigma\left(\langle \mathbf{w}, \mathbf{x}_i \rangle y_i\right)\right) + const(\mathbf{w})$$

и задача сводится к его максимизации и нахождению неизвестного вектора параметров $\mathbf{w}$.

$$L\left(\mathbf{w}, \mathbf{X}\right) \to \max_{\mathbf{w}}.$$

Задача максимизации $L(\mathbf{w}, \mathbf{X})$ эквивалентна минимизации эмпирического риска

$$\mathbf{Q}(\mathbf{w}, \mathbf{X}) = \sum_{i=1}^{T-\Delta} \log\left(1 + \exp\left(-\langle \mathbf{w}, \mathbf{x} \rangle y_i\right)\right) \to \min_{\mathbf{w}}$$

Описание алгоритма

Для удобства перечислим вспомогательные обозначения, которые будут введены и использованы в дальнейшем:

P — число строк в матрице $\mathbf{X}$.

L — число признаков объекта, или иначе, количество столбцов матрицы $\mathbf{X}$.

m — число обектов, выбираемых из $\mathbf{X}$ для обучения.

λ — параметр шага градиентного спуска.

s — число раз восстановления регрессии и получения векторов $\mathbf{w}$ при фиксированном m, но различных разбиениях $\mathbf{X}$ и $\mathbf{y}$ на обучающие и конрольные выборки.

N — число временных рядов.

i_{max} — максимальное число шагов градиентного спуска.

δ — критерий остановки градиентного спуска

Для определенности будем прогнозировать тенденцию первого ряда. Делаем разметку, описанную в постановке задачи с одним исключением, если $a_{1,t+1} = a_{1,t}$ будем ставить вместо a_i не 0, а -1. Так как оптимальную глубину лагирования Δ экспертно задать затруднительно, проведем вычисления для различных Δ и возьмем наилучшую, сравнивая полученные результаты. Нам не интересны абсолютные значения остальных временных рядов и мы не знаем ничего о том, какой ряд влияет на интересующий нас сильнее, а какой слабее. Все их значения будут входить линейно в σ, поэтому разумно нормировать остальные ряды. Пусть a_i^{max} — максимальное значения ряда a_i. Тогда новые значения $a_{i,t}^n$ нормированных рядов будут:

$$a_{i,t}^n = \frac{a_{i,t}}{a_i^{max}}.$$

Теперь имеем пучок рядов, где первый ряд размеченный, остальные нормированны. Зафиксировав Δ, составляем матрицу признаков $\mathbf{X}$ и столбец ответов к ней $\mathbf{y}$:

$$\mathbf{X}_t = \begin{pmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,L-1} & x_{1,L} \\ x_{2,1} & x_{2,2} & \dots & x_{2,L-1} & x_{2,L} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{P,1} & x_{P,2} & \dots & x_{P,L-1} & x_{T,L} \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_P \end{pmatrix}$$

где $P = T - \Delta$ — число получившихся строк признаков, а $L = N * \Delta$ — число признаков объекта y_i. Теперь разобьем матрицу $\mathbf{X}$ и столбец $\mathbf{y}$ на 2 части: обучающую и контрольную. Случайным образом выберем m строк из матрицы $\mathbf{X}$ и соответствующие этим строкам значения из столбца $\mathbf{y}$. Составим из них обучающую выборку, т.е матрицу и столбец $\mathbf{X}^{ed}, y^{ed}$. Из оставшихся составим $\mathbf{X}^{ch}, y^{ch}$ — контрольную выборку. Число m тоже является параметром и интерестно посмотреть, при каком процентном соотношении между m и общим числом объектов P мы будем получать оптимальный результат. Далее будет описан метод восстановления регрессионной модели методом градиентного спуска [2].

В качестве начального приближения зададим вектор весов модели как нулевой вектор:

$$\mathbf{w} = \begin{pmatrix} w_1 & w_2 & \dots & w_L \end{pmatrix}^T, \quad w_i = 0, \quad i \in \{1, 2, \dots, L\}$$

Тогда для шага k формула имеет вид:

$$\mathbf{w^{(k+1)}} = \mathbf{w^{(k)}} - \lambda \nabla \mathbf{Q}(\mathbf{w^{(k)}}, \mathbf{X}),$$

где λ есть параметр, влияющий на величину градиентного шага, а $\mathbf{Q}(\mathbf{w}, \mathbf{X})$ — эмпирический риск. В данной работе λ это достаточно малая константа, но для оптимизации скорости сходимости ее можно выбирать, например, по правилу Армихо [2]. Посчитаем производную сигмоидной функции:

$$\sigma'(z) = \frac{d}{dz}\ \frac{1}{1+\exp(-z)} = \frac{1}{1+\exp(-z)}\left(\frac{\exp(-z)}{1+\exp(-z)}\right) = \sigma(z)\sigma(-z).$$

С учетом этого вектор градиента функционала записывается как:

$$\triangledown \mathbf{Q}(\mathbf{w}, \mathbf{X}) = -\sum_{i=1}^{m} y_i \mathbf{x_i} \sigma\left(M_i(\mathbf{w})\right),$$

и градиентный шаг будет иметь вид:

$$\mathbf{w^{(k+1)}} = \mathbf{w^{(k)}} - \lambda \sum_{i=1}^{m} y_i \mathbf{x_i} \sigma\left(M_i(\mathbf{w}^{(k)})\right).$$

В качестве критерия остановки градиентного спуска будем использовать минимальное значение разности эмпирического риска на k-ом и $k+1$-ом шаге — δ. Если изменение значения эмпирического риска на протяжении нескольких итераций изменялось меньше чем на δ, то будем считать, что мы нашли оптимальный вектор весов $\mathbf{w}$. На случай, если значение эмпирического риска будет колебаться возле значения с амплитудой большей, чем заданная δ, введем максимально допустимое количество итераций i_{max}. Для предотвращения переобучения будем проверять значение эмпирического риска, если на протяжении нескольких итераций оно будет возрастать, то останавливаемся.

Полученный вектор весов проверяем на $\mathbf{X}^{ch}$ и $\mathbf{y}^{ch}$. В качестве способа оценки качества полученного вектора весов $\mathbf{w}$ можно использовать, например, площадь под ROC кривой, построенной на контрольной выборке или количество ошибок, допущенных на контрольной выборке. В данной работе используется последний вариант. Так как мы разбиваем матрицу $\mathbf{X}$ и столбец $\mathbf{y}$ случайным образом, проделаем эту процедуру s раз и выберем наилучший вектор весов $\mathbf{w}$ по описанному выше критерию.

Проверка на синтетических данных

Проверим метод логистической регрессии на синтетических данных, например на зашумленных синусах и периодических зашумленных трапециях. На входе имеются $N = 7$ схожих временных рядов вида произведения синусов (см. рис.1), или трапеций (см. рис.2) одинаковой длины $T = 100$ и с одинаковым временным шагом $t = 1$.

Восстановим регрессию, задав следующие параметры: $i_{max} = 1000$, $\lambda = 0.005$, $m = 70$, $\delta = 0.001$, $s = 20$. Проверим эффективность алгоритма при разной глубине лагирования Δ.

В левом столбце приведены графики для синусов, в правом для трапеций. На рис.3 и рис.4 иллюстрируется сходимость метода. По оси ординат откладывается значение эмпирического риска, а по оси абсцис — номер шага градиентного спуска. Как видно из графиков, метод градиентного спуска сходится монотонно.

Далее приведены таблицы с процентными соотношениями числа ошибок к общему числу объектов и значениями AUC для ROC кривых для различных Δ (см. таб. 1 и 2). Так же приводятся ROC кривые для разного значения параметра Δ (см. рис.5 – 12). Синим

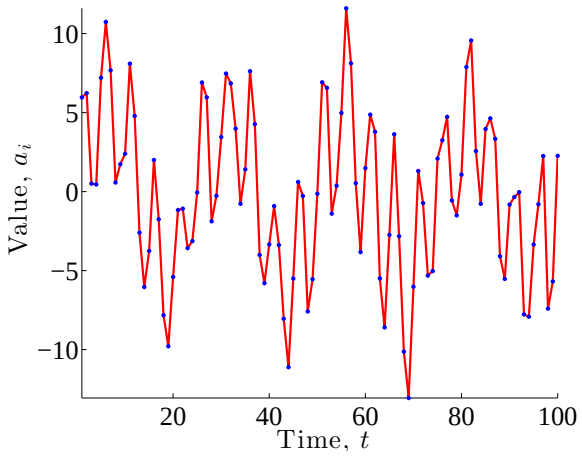


Рис. 1. Пример ряда — синус.

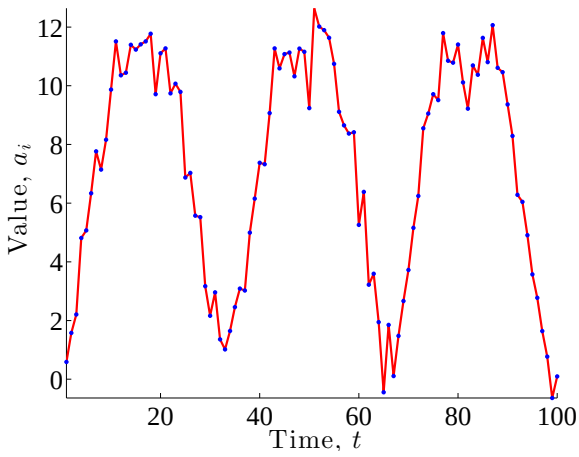


Рис. 2. Пример ряда — трапеция.

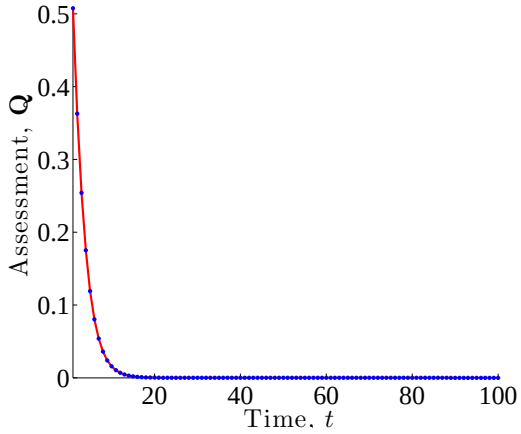


Рис. 3. Зависимость **Q** от номера градиентного шага, синусы.

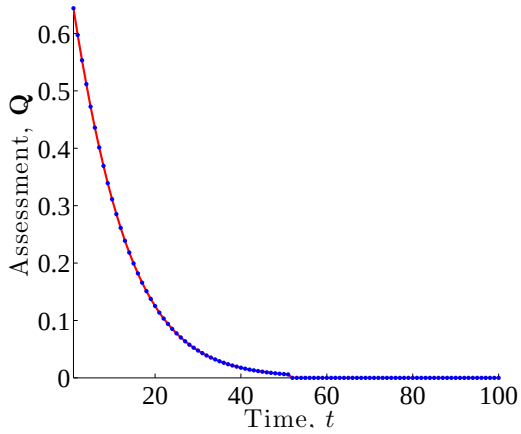


Рис. 4. Зависимость **Q** от номера градиентного шага, трапеции.

цветом отображаюстя ROC кривые, построенные по контрольной выборке, красным — по обучающей. Зеленым цветом проведена кривая, соответствующая случайному предсказанию (исходы $+1$ и -1 полагаются равновероятными).

Δ	процент ошибок	AUC для обучающей выборки	AUC для контрольной выборки
2	21	0.8253	0.8773
4	16	0.8962	0.9026
6	12	0.9676	0.9697
8	13	0,9588	0.9388

Таблица 1. Результаты для произведения синусов

Как можно видеть из результатов, при увеличении параметра Δ качество модели заметно улучшается, затем при некотором значении Δ_{opt} мы имеем минимальный процент ошибок, максимальное значение AUC для контрольной выборки и достаточно высокое

значение AUC для обучающей. При дальнейшем росте Δ количество ошибок начинает увеличиваться, а значение AUC для контрольной выборки уменьшаться. Тем не менее значение AUC для обучающей выборки продолжает расти. Это объясняется тем, что при увеличении Δ мы увеличиваем количество признаков, и поэтому точнее можем обучиться. Но после экстремального значения Δ_{opt} мы начинаем переобучаться и поэтому на контрольной выборке делаем больше ошибок. Показатели AUC на контрольных выборках при $\Delta \leqslant \Delta_{opt}$ оказываются явно лучше, чем на обучающих. Это связано с тем, что мы выбираем тот вектор весов $\mathbf{w}$, при котором мы делаем наименьшее число ошибок на контрольной выборке.

Δ	процент ошибок	AUC для обучающей выборки	AUC для контрольной выборки
2	39	0.6669	0.7222
4	25	0.7239	0.7306
6	38	0.7108	0.7048
8	42	0,7680	0.5865

Таблица 2. Результаты для трапеций

Стоит отметить, что для синусоидальных данных алгоритм работает на порядок лучше. Сравним результаты (см. таб. 3) при оптимальном значении $\Delta_{opt} = 8$ для синусов и $\Delta_{opt} = 6$ для трапеций. Обучаясь по 70% объектам, мы имеем процент ошибок для трапеций вдвое больший, чем для синусов, а значения AUC явно ниже.

тип ряда	процент ошибок	AUC на обучении	AUC на контроле
синусы	12	0.9676	0.9697
трапеции	25	0.7239	0.7306

Таблица 3. Сравнение результатов

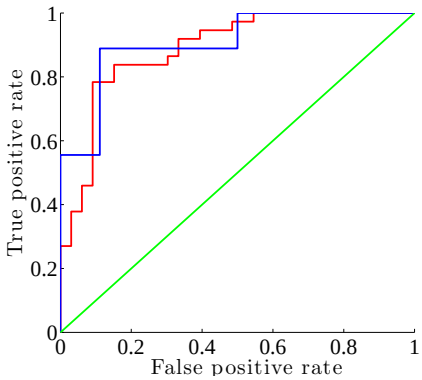


Рис. 5. ROC кривая, синусы, $\Delta = 2$.

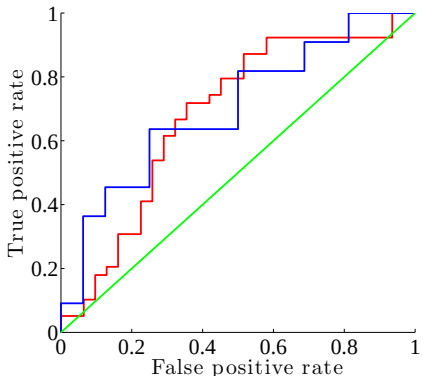


Рис. 6. ROC кривая, трапеции, $\Delta = 2$.

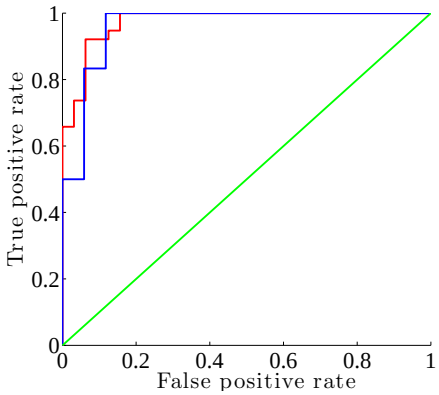


Рис. 7. ROC кривая, синусы, $\Delta = 6$.

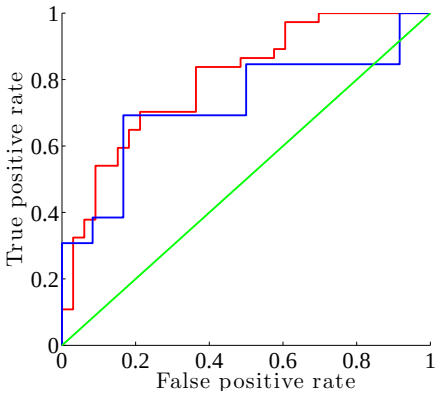


Рис. 8. ROC кривая, трапеции, $\Delta = 4$.

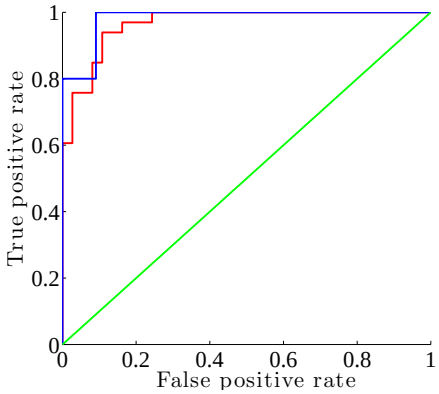


Рис. 9. ROC кривая, синусы, $\Delta = 8$.

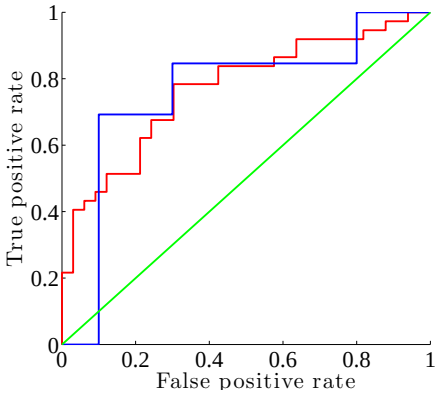


Рис. 10. ROC кривая, трапеции, $\Delta = 6$.

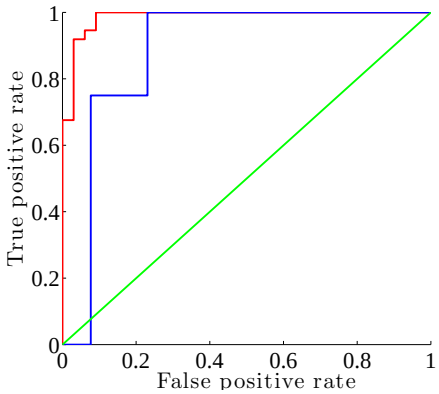


Рис. 11. ROC кривая, синусы, $\Delta = 12$.

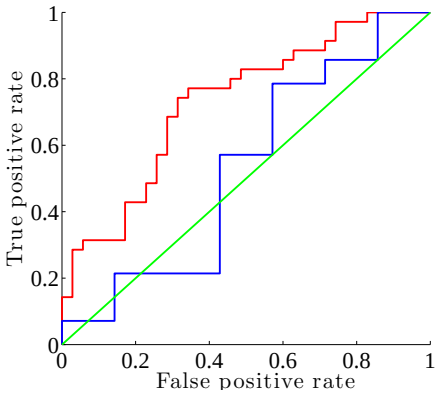


Рис. 12. ROC кривая, трапеции, $\Delta = 8$.

Прогноз потребления электроэнергии

Проверим работоспособность алгоритма на данных о потреблениии электроэнергии с 01.01.08 по 25.04.08. Данные имеют вид, приведенный в таб.4. Количество строк в таб-

лице — 2767. Связь между этими временными рядами явно прослеживается. Например, количество потребляемой энергии явно зависит от температуры, так как при низкой температуре люди начинают использовать электронагреватели, а при высокой — кондиционеры, потребляемая мощность которых довольно высокая. От времени суток зависит количество электроэнергии, тратящееся на освещения, а от дня недели — количество людей на работе и дома, что в свою очередь влияет на количество работающей аппаратуры. Исходя из всего вышеперечисленного, будем считать, что все эти ряды довольно сильно коррелируют между собой. Поэтому целесообразно исследовать их всех вместе, как временной пучок. Каждая строчка в наших обозначениях будет соответствовать интервалу $t = 1$, тогда длина этих рядов будет $T = 2767$. Приведем зависимость значения потребления электроэнергии от времени (см. рис. 14) для небольшого временного интервала.

потребление электроэнергии в МВт*ч	дата	день недели	час	температура
1366,74115	01.01.08	2	0:00	-11,9
1333,16888	01.01.08	2	1:00	-12,0
1293,20544	01.01.08	2	2:00	-12,0
1302,07739	01.01.08	2	3:00	-12,0
...	...	...	...	...

Таблица 4. Вид данных о потреблении электроэнергии

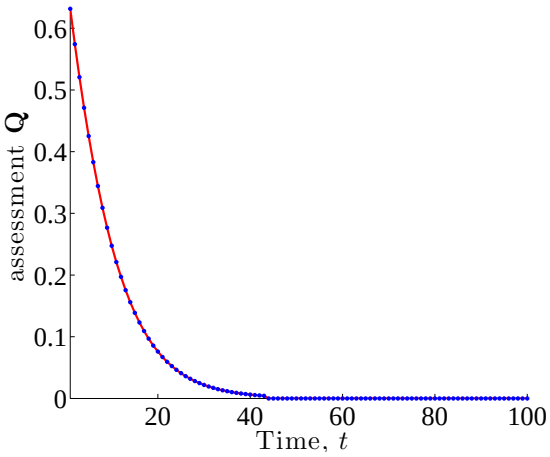


Рис. 13. Зависимость **Q** от номера градиентного шага, синусы.

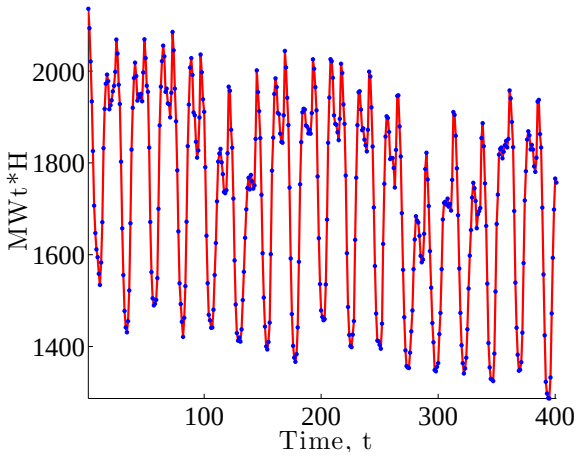


Рис. 14. Зависимость потребления электроэнергии от времени

Как можно заметить из рис.14, данные имеют вид синусоид. Поэтому основываясь на результатах для синтетических синусоид, можно предположить, что результаты для этих данных также будут хорошими.

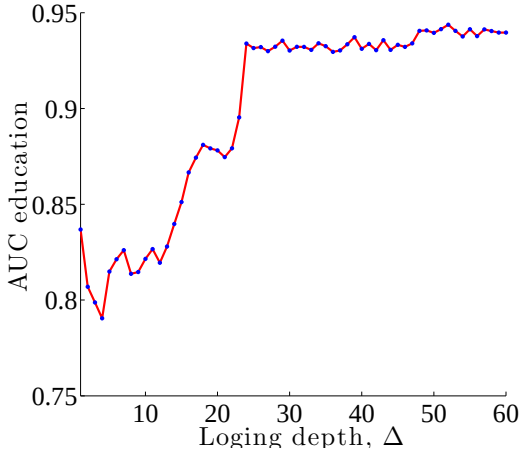


Рис. 15. Зависимость AUC на обучающей выборке от Δ.

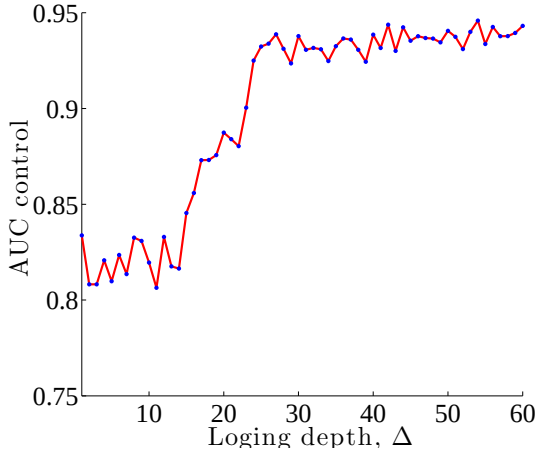


Рис. 16. Зависимость AUC на контрольной выборке от Δ.

Мы будем прогнозировать тенденцию потребления электроэнергии (увеличится оно или уменьшится в следующий момент времени) т.е. первый ряд таблицы. Как и для синтетических данных, размечаем его, а затем строим матрицу $\mathbf{X}$ и столбец $\mathbf{y}$. Восстановим регрессию, задав следующие параметры: $i_{max} = 1000$, $\lambda = 0.001$, $m = 70\%$, $\delta = 0.001$, $s = 10$. Величину Δ будем изменять от 1 до 60, с шагом 1 и найдем оптимальное значение для наших данных. Оценивать качество модели при заданном Δ будем по трем параметрам: AUC для ROC кривой на обучающей выборке, AUC для ROC кривой на контрольной выборке и процент ошибок от общего числа объектов. На рисунках (16, 15, 17) показаны зависимости этих параметров от Δ. Можно заметить, что начиная с $\Delta = 24$ значения выходят на плато и практически не изменяются. Число ошибок медленно уменьшается, но это скорее связано с переобученностью, чем с улучшением качества алгоритма. А так как при увеличении Δ время восстановления и объем обрабатываемых данных сильно увеличивается, разумно взять $\Delta_{opt} = 25$, т.к. дальнейшее увеличение не даст нам значительного выигрыша.

Значение $\Delta_{opt} = 24$ можно оъяснить и с логической точки зрения. Так как в сутках 24 часа (а данные повторяются периодически с периодом в сутки), шаг по времени наших данных есть $t = 1$час, то наиболее полную информацию о таком ряде мы будем получать, зная его предысторию за предшествующий период, т.е 24 часа или 24 шага по времени.

Δ	процент ошибок	AUC для обучающей выборки	AUC для контрольной выборки
25	14.1	0.9276	0.9398

Таблица 5. Результаты прогноза потребления энергии для Δ_{opt}

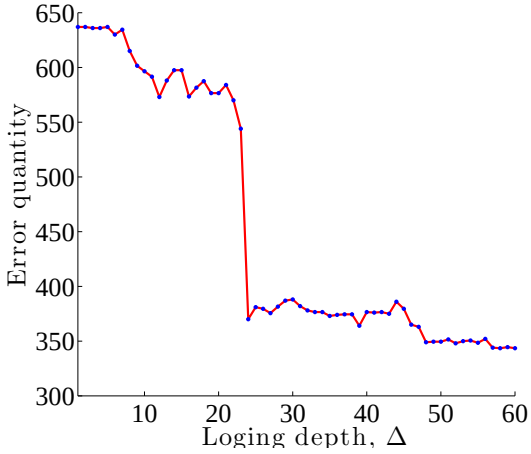


Рис. 17. Зависимость количества ошибок на данных от Δ.

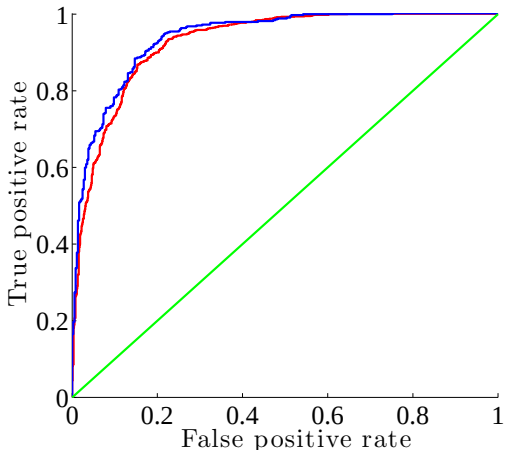


Рис. 18. ROC кривые при оптимальном Δ.

Литература

[1] *К.В.Воронцов.* Лекции по линейным алгоритмам классификации. / К.В.Воронцов.

[2] *В.Г.Жадан.* Численные методы решения задач оптимизации / В.Г.Жадан.

[3] *Н.В.Филипенков.* — О задачах анализа пучков временных рядов с изменяющимися закономерностями. — Master's thesis, 2006.

[4] *Н.В.Филипенков.* Об алгоритмах прогнозирования процессов с плавно меняющимися закономерностями: Ph.D. thesis. — 2010.

[5] *Б.А.Романенко.* Событийное моделирование и прогноз финансовых временных рядов / Б.А.Романенко. — 2011.

[6] *Ng, A.* Classification and logistic regression / A. Ng.

Прогноз квазипериодических временных рядов непараметрическими методами*

Е. Ю. Клочков
eklochov@gmail.com
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В работе рассматривается непараметрический метод прогнозирования квазипериодических временных рядов. В качестве метода используется квантильная регрессия. Его преимущества в том что, несмотря на его простоту, он хорошо приближает многие из известных распределений. Предлагаемый метод тестируется на данных о продажах продуктов.

Ключевые слова: *квантиль, квантильная регрессия, линейное программирование.*

Введение

Использование квантильной регрессии началось с так называемых «неэффективных статистик» с использованием нескольких квантилей для замены «на скорую руку» [6]. Позднее, Р. Коенкер и Г. Бассет-младший расширили понятие обычных квантилей в модели локализации до более общего класса линейных моделей, в которых условные квантили имеют линейную форму [4].

Мы называем число $\theta-$квантилью случайной величины, если с вероятностью θ случайная величина не превосходит это число. Например, $\frac{1}{2}-$ квантиль есть ничто иное как медиана. Было обнаружено что взвешенное среднее арифметическое $\frac{1}{4}$, $\frac{1}{2}$, $\frac{3}{4}$ квантилей с коэффициентами 0.3, 0.4, 0.3 имеет асимптотическую эффективность почти восемьдесят процентов для нормального, лапласова, логистического распределений и распределения Коши [3]. Тем самым, отвлекаясь от конкретного вида распределения, квантильная регрессия захватывает широкий класс задач.

В работе [2] для прогнозирования временного ряда применяется медианная регрессия. В предположени, что значение временного ряда зависит от нескольких предшествующих ему, рассматриваемых в качестве признака, мы пронаблюдаем как с возрастанием числа признаков изменяется ошибка. Настоящий метод также применяется для расчета компенсации работникам [5], оценивания нетто-премий страховыми компаниями [1] и других задач.

Постановка задачи

Имеется выборка $\{y_i, \mathbf{x}_i\}_{i=1}^t$, где $\mathbf{x}_i$ — $k \times 1$ вектор независимых признаков. Мы предполагаем, что условные квантили определяются соотношением

$$\mathrm{Quant}_\theta(y_i|\mathbf{x}_i) = \mathbf{x}_i^T\mathbf{b} + \varepsilon_i. \tag{1}$$

Оценка $\mathbf{b}_\theta$ определяется из соотношения

$$\hat{\mathbf{b}}_\theta = \arg\min_{\mathbf{b}\in\mathbb{R}^k} \sum_{i:y_i\geqslant\mathbf{x}_i^T\mathbf{b}} \theta|y_i - \mathbf{x}_i^T\mathbf{b}| + \sum_{i:y_i<\mathbf{x}_i^T\mathbf{b}} (1-\theta)|y_i - \mathbf{x}_i^T\mathbf{b}|. \tag{2}$$

Линейная модель (2) была представлена Коенкером и Бассетом [4], как обобщение простой квантили.

Научный руководитель В. В. Стрижов

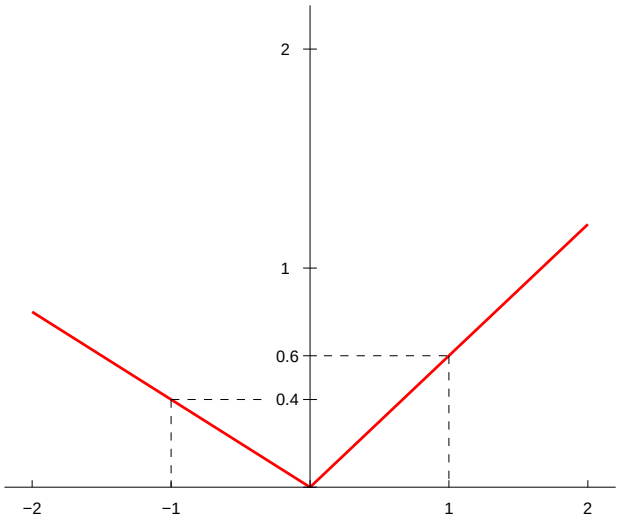


Рис. 1. График функции $\rho_{0.6}(y)$.

Определение 1. *Пусть задана случайная величина ξ. Тогда для $\theta \in (0,1)$ назовем θ-квантилью этой случайной велечины такое число x_θ, что*

$$\begin{cases} \mathsf{P}(\xi \leqslant x_\theta) \geqslant \theta, \\ \mathsf{P}(\xi < x_\theta) \leqslant \theta. \end{cases} \tag{3}$$

Как известно, медиана обладает наименьшим математическим ожиданием модуля отклонения. Аналогично можно показать, что при произвольном θ, положив функцию потери $\rho_\theta(y) \triangleq y(\theta - [y < 0])$, соответствующая квантиль находится решением задачи

$$\arg\min_{u \in \mathbb{R}} \mathsf{M}\rho_\theta(\xi - u). \tag{4}$$

Это можно легко проверить, продифференцировав $\mathsf{M}\rho_\theta(\xi - u)$ по u. Действительно, распишем выражение $\mathsf{M}\rho_\theta(\xi - y)$ в виде суммы двух интегралов

$$\mathsf{M}\rho_\theta(\xi - u) = (1-\theta)\int_{-\infty}^{u}(u-y)dF_\xi(y) +$$

$$+\theta\int_{u}^{\infty}(y-u)dF_\xi(y)$$

и продифференцируем по u. Получим

$$(1-\theta)\int_{-\infty}^{u} dF_\xi(y) - \theta\int_{u}^{\infty} dF_\xi(y) = 0,$$

откуда,

$$(1-\theta)F_\xi(u) - \theta(1 - F_\xi(u)) = 0,$$

$$F_\xi(u) = \theta,$$

что и соответствует определению квантили в случае, когда F непрерывная строго возрастающая.

При прогнозировании искомое значение y сопоставляется со случайной величиной ξ. Тогда для конечного набора значений выражение (4) переписывается в виде

$$\arg\min_{u \in \mathbb{R}} \sum_{i=1}^{t} \rho_\theta(y_i - u). \tag{5}$$

Эта задача обобщается до описаной выше в случае, когда ищется условная квантиль.

Представление в виде задачи ЛП

Задача (2) сводится к линейному программированию. У нас есть обучающая выборка $(y_i, \mathbf{x}_i)_{i=1}^t$. Из признаков составим матрицу $X = ||\mathbf{x}_1\, \mathbf{x}_2\, \ldots\, \mathbf{x}_t||$. Положим

$$\mathbf{r} = y - X^T\mathbf{b}$$

тогда,

$$r_i = y_i - \mathbf{x}_i^T\mathbf{b},\ i = \overline{1,\, t}.$$

Перепишем задачу (2) в виде

$$\hat{\mathbf{b}}_\theta = \arg\min_{\mathbf{b}\in\mathbb{R}^k} \sum_{i=1}^{t} \theta(y_i - \mathbf{x}_i^T\mathbf{b})_+ + (1-\theta)(y_i - \mathbf{x}_i^T\mathbf{b})_-,$$

где $(\cdot)_+$ и $(\cdot)_-$ — положительная и отрицательная части числа, соответственно. Тогда положив $\mathbf{r} = \mathbf{r}^+ - \mathbf{r}^-$, где $\mathbf{r}^+$, $\mathbf{r}^-$ – положительная и отрицательная части вектора $\mathbf{r}$, получаем задачу ЛП

$$\begin{cases} \theta\mathbf{e}^T\mathbf{r}^+ + (1-\theta)\mathbf{e}^T\mathbf{r}^- \to \min \\ X^T\mathbf{b} + \mathbf{r}^+ - \mathbf{r}^- = \mathbf{y} \\ (\mathbf{b}, \mathbf{r}^+, \mathbf{r}^-) \in \mathbb{R}^k \times {\mathbb{R}_+}^{2t}, \end{cases} \tag{6}$$

где $\mathbf{e} = (1, 1, \ldots, 1)^T$. Легко понять, что ${r^+}_i$ и ${r^-}_i$ не могут иметь ненулевые значения одновременно, поэтому задачи (2) и (6) эквивалентны. На выходе мы получаем решение $(\mathbf{b}, \mathbf{r}^+, \mathbf{r}^-)$, из которого нам нужен вектор $\mathbf{b}$.

Задачу также можно представить в каноническом виде

$$\begin{cases} \theta\mathbf{e}^T\mathbf{r}^+ + (1-\theta)\mathbf{e}^T\mathbf{r}^- \to \min \\ X^T\mathbf{b}^+ - X^T\mathbf{b}^- + \mathbf{r}^+ - \mathbf{r}^- = \mathbf{y} \\ (\mathbf{b}^+, \mathbf{b}^-, \mathbf{r}^+, \mathbf{r}^-) \in {\mathbb{R}_+}^{2k+2t}, \end{cases} \tag{7}$$

что на практике ускоряет работу алгоритма. Здесь в качестве решения задачи (2) возвращаем $\mathbf{b} = \mathbf{b}^+ - \mathbf{b}^-$.

Прогнозирование временных рядов

При прогнозировании временных рядов мы предполагаем, что каждое следующее значение временного ряда зависит от предыдущих. Тем самым в качестве признаков мы выбираем k предыдущих значений ряда: дан ряд $\{y_i\}_{i=1}^t$ — мы строим выборку $\{y_i, \mathbf{x}_i\}_{i=k+1}^t$, где

$$\mathbf{x}_i = (y_{i-1}, \ldots, y_{i-k})^T. \tag{8}$$

Решение представляется в виде

$$y_\theta = b_1 y_t + b_2 y_{t-1} + \ldots + b_k y_{t-k+1} \tag{9}$$

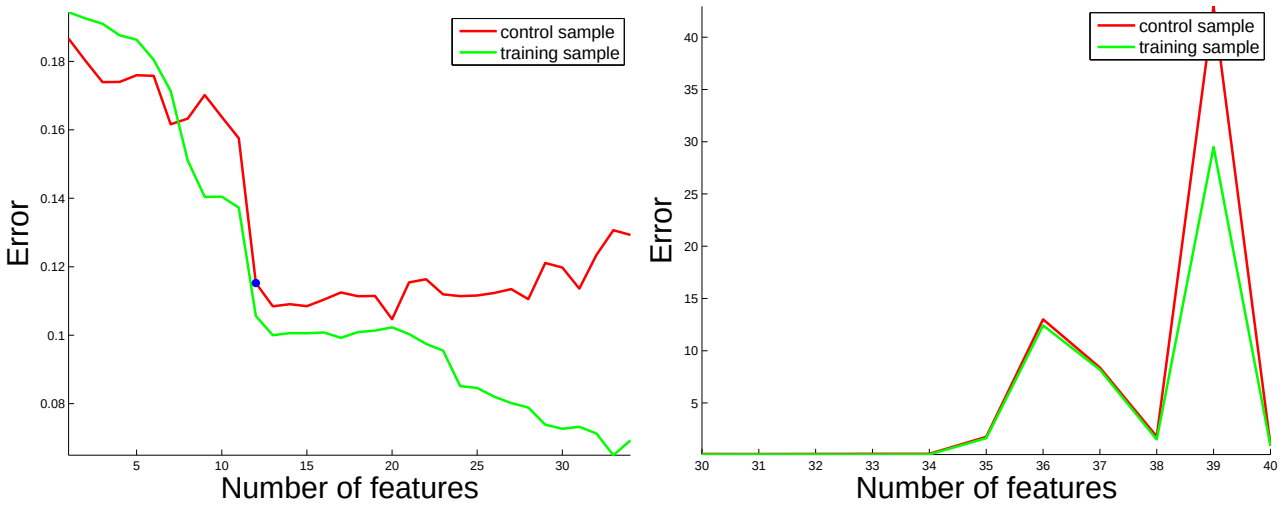
Так же может оказаться полезным положить в качестве $k+1$ -го признака единицу. Тогда решением будет

$$y_\theta = b_0 + b_1 y_t + b_2 y_{t-1} + \ldots + b_k y_{t-k+1} \tag{10}$$

В качестве функции потерь будем использовать $L(\tilde{y}, y) = |\tilde{y} - y|$, где $\tilde{y}$ – предсказание, y – известное значение выборки. Как уже отмечалось, наименьшим отклонением суммы модулей обладает медиана, поэтому в качестве предсказания будем брать $\tilde{y}_i = \mathbf{x}_i^T \mathbf{b}_{0.5}$.

Вычислительный эксперимент

Для вычислительного эксперимента используются данные о продажах красных вин в Австралии по месяцам с января 1980 по июнь 1994. Данные можно считать приближенно периодическими с периодом в 12 месяцев.



(a) (b)

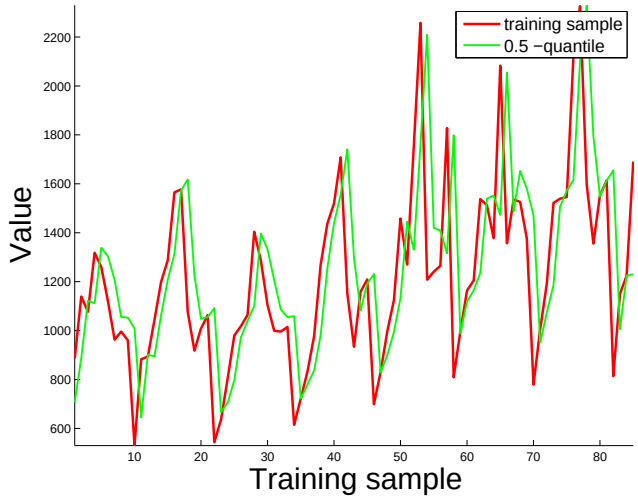
Рис. 2. Относительная ошибка на медиане, в зависимости от числа признаков.

Выборка, описанная в (8) разделена на две равные части — обучающая и контрольная. Обучающая используется для вычислений, контрольная — для оценки точности модели. На рис. 2 показана зависимость относительной ошибки от числа значений временного ряда, выбираемых в качестве признаков. Можно увидеть резкий спад на 12-ти признаках (точка выделена синим цветом), что соответствует одному году. На рис. 3 и 4 показаны результаты вычислений на обучающей и контрольной выборках, соответственно. По оси абсцисс откладываются месяцы, по оси ординат величины продаж, соответствующие месяцам. Полученные результаты сравниваются с известными значениями выборки при различных числах признаков k. Из графиков видно, что при $k = 3$ и 5 спады и подъемы сдвинуты, при $k = 12$ этот эффект пропадает. Коэффициенты вектора $\mathbf{b}$ для этого случая приведены в таблице 1.

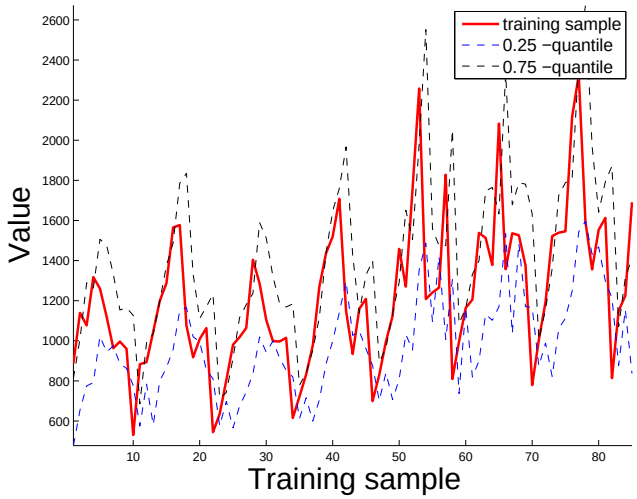
θ	1	2	3	4	5	6	7	8	9	10	11	12
0.25	0.52	-0.08	0.19	0.24	0.11	-0.11	-0.18	-0.03	-0.02	0.23	0.02	0.05
0.5	0.63	0.05	0.2	0.07	0.03	-0.1	-0.07	0.01	0.01	0.15	0.07	0.01
0.75	0.82	0.09	0.02	0.01	0.14	-0.09	-0.06	-0.03	0.09	0.00	0.09	0.08

Таблица 1. Векторы $\mathbf{b}_{0.25}, \mathbf{b}_{0.5}, \mathbf{b}_{0.75}$

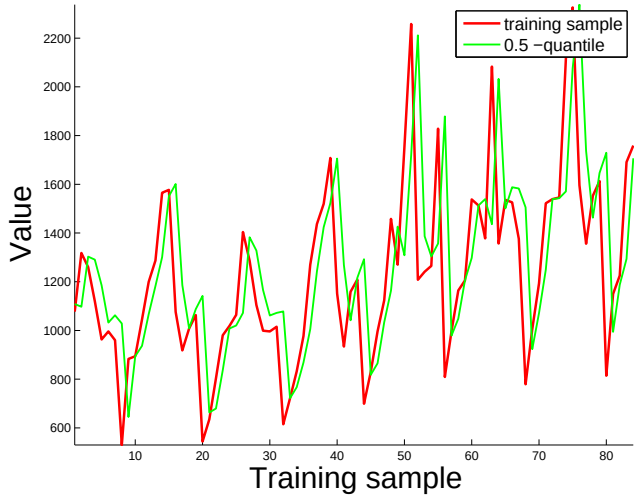
Как видно, значительно больше коэффициент при признаке, соответствующем значению ряда предыдущего года того же месяца. Тем не менее, квантили с меньшим числом



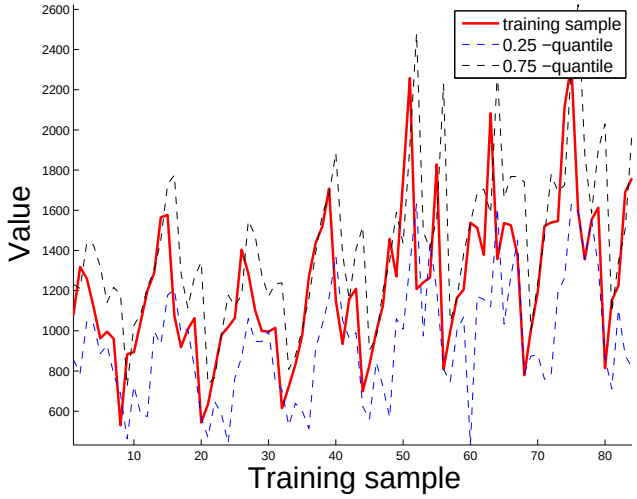
(a) $\theta = 0.5 \quad k = 3$



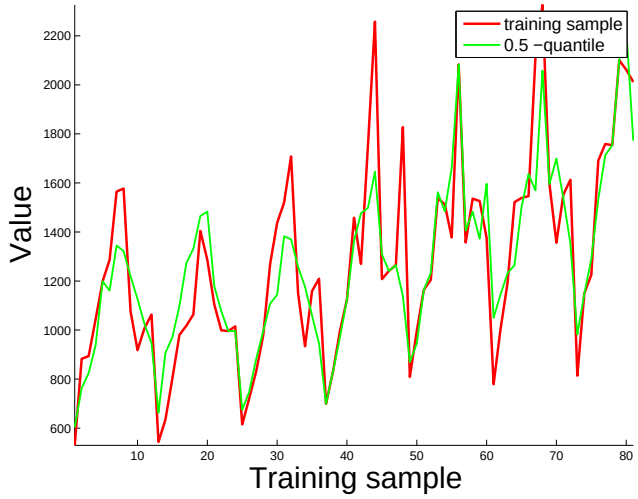
(b) $\theta = 0.25,\ 0.75 \quad k = 3$



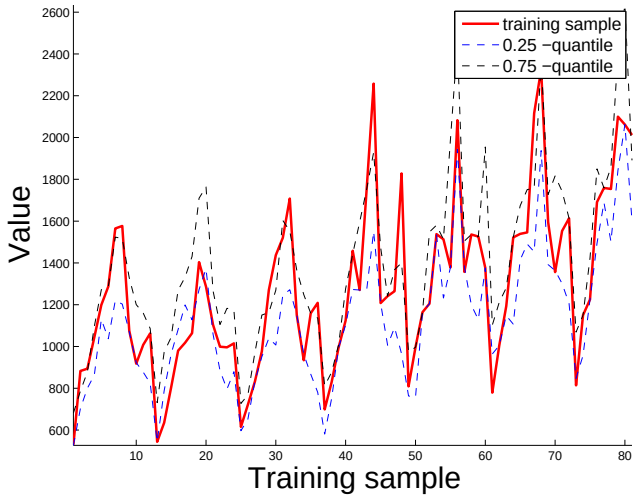
(c) $\theta = 0.5 \quad k = 5$



(d) $\theta = 0.25,\ 0.75 \quad k = 5$

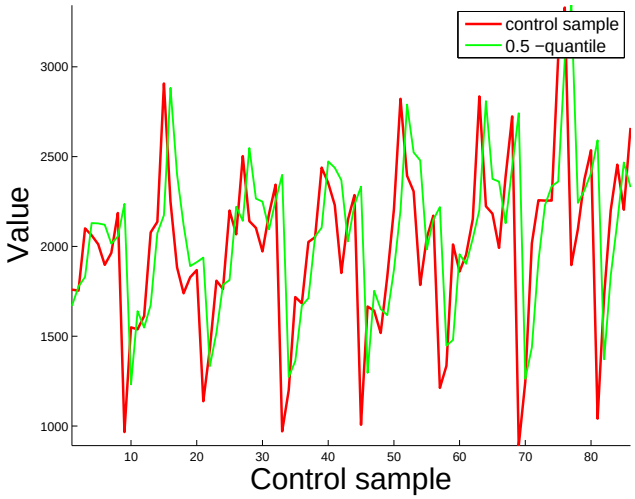


(e) $\theta = 0.5 \quad k = 12$

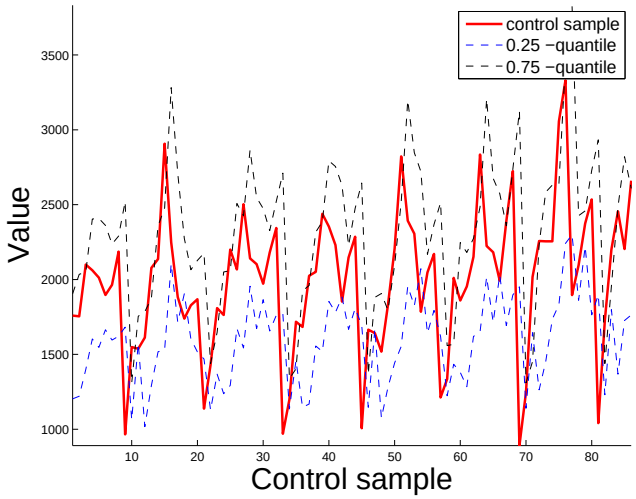


(f) $\theta = 0.25,\ 0.75 \quad k = 12$

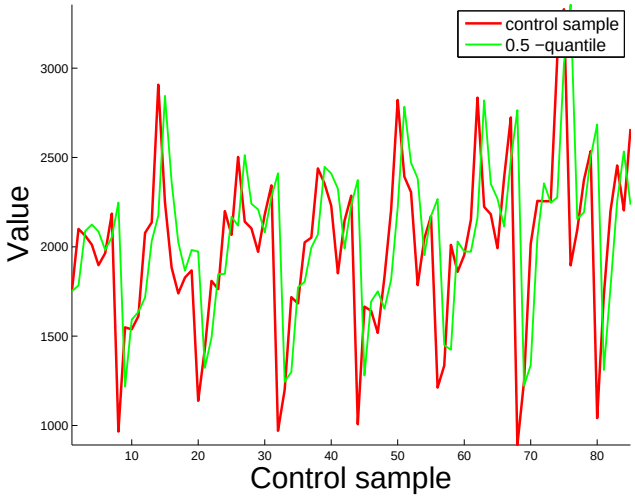
Рис. 3. Медиана и 0.25- и 0.75- квантили на обучающей выборке, число признаков $k = 3,\ 5,\ 12$



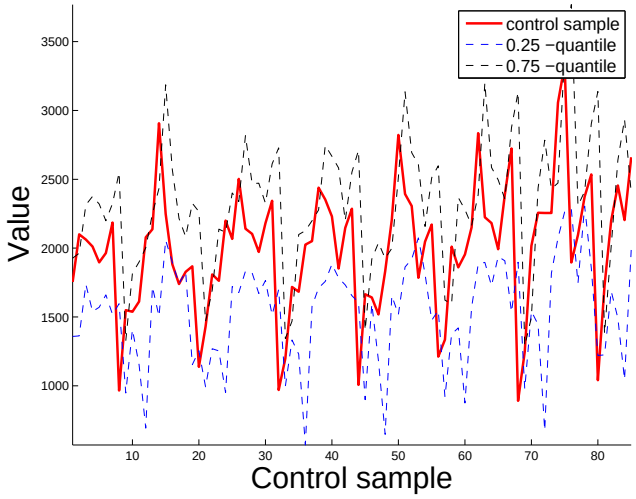
(a) $\theta = 0.5 \quad k = 3$



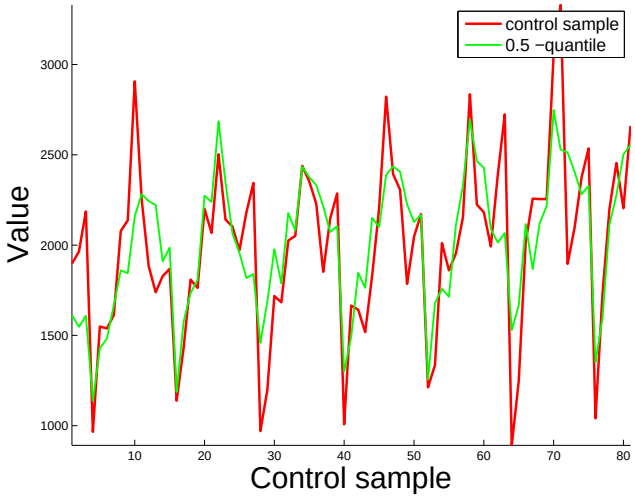
(b) $\theta = 0.25,\ 0.75 \quad k = 3$



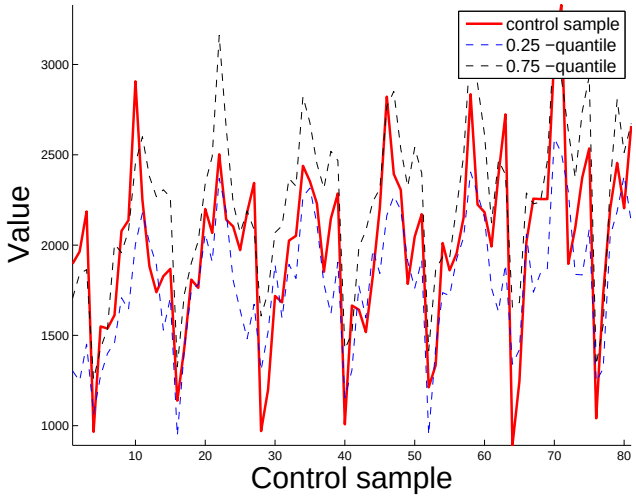
(c) $\theta = 0.5 \quad k = 5$



(d) $\theta = 0.25,\ 0.75 \quad k = 5$



(e) $\theta = 0.5 \quad k = 12$



(f) $\theta = 0.25,\ 0.75 \quad k = 12$

Рис. 4. Медиана и 0.25- и 0.75- квантили на контрольной выборке, число признаков $k = 3,\ 5,\ 12$

признаков также хорошо приближают искомое значение — как видно из рис. 2, на 3-ех, 5-ти и 12-ти признаках ошибки соответственно, 0.19, 0.16 и 0.11 . На рис. 2(b) показана ошибка при k от 30 до 40. На графике видно, что при большом числе k ошибка сильно возрастает как на контрольной выборке (control sample), так и на обучающей (training sample). Поэтому это связано не с переобучением, а с тем, что метод оптимизации (7) расходится при больших входных данных.

Заключение

В работе рассматривалась задача прогнозирования временного ряда с использованием квантильной регрессии. Эксперимент показал, что даже при небольшом числе признаков метод дает хорошие результаты. Кроме того было отмечено, что при наблюдении за ошибкой при варьировании числа признаков, происходит резкий спад на периоде квазипериодического ряда. В ходе эксперимента переобучение не наблюдалось.

Литература

[1] *Абдурманов, Р.* Применение квантильной регрессии для оценки нетто-премий / Р. Абдурманов // *Ломоносов.* — 2008.

[2] *Литвинов, И.* Прогнозирование объемов продаж новых товаров (отчет) / И. Литвинов // *MachineLearning.ru.* — 2009.

[3] *Постникова, Е.* — Квантильная регрессия. — Master's thesis, НГУ, 2000.

[4] *Koenker, R.* Quantile regression / R. Koenker, J. Gilbert Bassett // *Econometrics.* — 1978. — Vol. 46, no. 1. — Pp. 33–50.

[5] *Koenker, R.* Quantile regression / R. Koenker, K. F. Hallock // *Journal of Economics Prospects.* — 2001. — Vol. 15, no. 4. — Pp. 143–156.

[6] *Mosteller, F.* On some useful "inefficient" statistics / F. Mosteller // *The Annals of Mathematical Statistics.* — Dec, 1946. — Vol. 17, no. 4. — Pp. 377–408.

Последовательный выбор признаков при восстановлении регрессии*

Л. Н. Леонтьева
liubov.sanduleanu@gmail.com
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

Исследуется проблема оптимальной сложности модели в связи с ее точностью и устойчивостью. Задача состоит в нахождении наиболее информативного набора признаков в условиях их высокой мультиколлинеарности. Для выбора оптимальной модели используется модифицированный алгоритм шаговой регрессии, являющийся одним из алгоритмов добавления и удаления признаков. Для описания работы пошагового алгоритма предложена модель n-мерного куба. Проанализированы величины матожидания и дисперсии функции ошибки.

Ключевые слова: *отбор признаков, мультиколлинеарность, шаговая регрессия, метод Белсли, прогнозирование временных рядов.*

Введение

Решается задача восстановления линейной регрессии при наличии большого числа мультиколлинеарных признаков. Термин «мультиколлинеарность» введен Р. Фишером при рассмотрении линейных зависимостей между признаками [1]. Проблема состоит в том, что количество признаков значительно превосходит число зависимых переменных, то есть мы имеем дело с переопределенной матрицей. Для решения этот задачи необходимо исключить наиболее малоинформативные признаки. Для отбора признаков предлагается использовать модифицированный метод шаговой регрессии.

Ранее для решения подобных задач использовались следующие методы: метод наименьших углов LARS [2], Лассо [3], ступенчатая регрессия [4], последовательное добавление признаков с ортогонализацией FOS [5, 6], шаговая регрессия [4, 7, 8] и другие [14]. Шаговыми методами называются методы, заключающиеся в последовательном удалении или добавлении признаков согласно определенному критерию. Существует несколько недостатков этих методов, например, важный признак может быть никогда не включен в модель, а второстепенные признаки будут включены.

В работе предложен модифицированный метод шаговой регрессии. Существует три основых разновидности шаговых методов: метод последовательного добавления признаков, метод последовательного удаления признаков и метод последовательного добавления и удаления признаков. В работе используется последний. Метод включает два основных шага: шаг Add (последовательное добавление признаков) и шаг Del (последовательное удаление признаков). Добавление признаков производится с помощью FOS [5, 6]. Данный метод последовательно добавляет признаки, которые максимально коррелируют с вектором регрессионных остатков. Удаление признаков в нашей работе осуществляется методом Белсли [9]. Он позволяет выявить мультиколлинеарность признаков, используя сингулярное разложение матрицы признаков. Для нахождения алгоритма, который доставляет одновременно точную и устойчивую, в смысле минимизации числа мультиколлиниарных признаков, модель предложен новый критерий останова этапов Add и Del, а так же останова всего алгоритма.

Научный руководитель В. В. Стрижов

Предложенный метод выбора модели проиллюстрирован задачей прогнозирования состояния здоровья людей больных диабетом. Ранее подобные задачи решались с помощью гребневой регрессии [10], метода наименьших углов, построения локальных регрессионных моделей [12, 13] и других.

Задача прогнозирования с помощью линейной регрессии

Даны временной ряд $\mathbf{s}^0 = \{x_i\}_{i=1}^{T-1}$, будем называть его целевым рядом, и временные ряды $\mathbf{s}^1, \mathbf{s}^2 \ldots \mathbf{s}^p$. Необходимо спрогнозировать следующее значение s_T^0 ряда $\mathbf{s}_0$.

Предполагается, что искомая величина s_T^0 зависит от последних b значений рядов $\mathbf{s}^0, \mathbf{s}^1 \ldots \mathbf{s}^p$. Параметр b называется глубиной логирования. Таким образом мы имеем дело с $b(p+1)$ признаком. Нахождение оптимальной модели, состоит в отыскании такого набора признаков, который минимизирует ошибку S при прогнозировании методом линейной регрессии.

Построим матрицу плана $\mathbf{X}^*$

$$\mathbf{X}^* = \left[\begin{array}{cccccccccc|c} s_0^1 & \ldots & s_0^b & s_1^1 & \ldots & s_1^b & \ldots & s_p^1 & \ldots & s_p^b & x_{b+1} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ s_0^{T-b-2} & \ldots & s_0^{T-2} & s_1^{T-b-2} & \ldots & s_1^{T-2} & \ldots & s_p^{T-b-2} & \ldots & s_p^{T-2} & x_{T-1} \\ \hline s_0^{T-b-1} & \ldots & s_0^{T-1} & s_1^{T-b-1} & \ldots & s_1^{T-1} & \ldots & s_p^{T-b-1} & \ldots & s_p^{T-1} & x_T \end{array}\right],$$

где s_i^j — j-ое значение ряда $\mathbf{s}_i$. То есть, строка с номером i матрицы плана X^* есть векторизованная подматрица, состоящая из значений временных рядов

$$\left[\begin{array}{ccc} s_0^i & \ldots & s_p^i \\ \ldots & \ddots & \ldots \\ s_0^{b-i} & \ldots & s_p^{b-i} \\ s_0^{b-i+1} & \ldots & s_p^{b-i+1} \end{array}\right].$$

Введем обозначения:

$$\mathbf{X}^* = \left[\begin{array}{c|c} \mathbf{X} & \mathbf{y} \\ \hline \mathbf{x}_m & x_T \end{array}\right].$$

Необходимо построить линейную регрессию:

$$\mathbf{y} = \mathbf{X}\mathbf{w}, \tag{1}$$

где $\mathbf{w}$ — вектор параметров. Тогда получим

$$x_T = \langle \mathbf{x}_m, \mathbf{w} \rangle .$$

Требуется решить задачу минимизации евклидовой нормы вектора регрессионных остатков

$$||\mathbf{X}\mathbf{w} - \mathbf{y}||^2 \to \mathbf{min}.$$

Вектор параметров $\mathbf{w}$ отыскивается с помощью метода наименьших квадратов

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{y}).$$

Задача выбора оптимальной модели

Опишем, в чем состоит задача выбора оптимальной модели. Задана выборка $D = (\{\mathbf{x}_i, y_i\}), i \in \mathcal{I}$, где множество свободных переменных — вектор $\mathbf{x} = [x_1, \ldots, x_j, \ldots, x_n]$,

проиндексированно $j \in \mathcal{J} = \{1, \ldots, n\}$. Задано разбиение множества индексов элементов выборки $\mathcal{I} = \mathcal{L} \sqcup \mathcal{C}$. Также задан класс линейных параметрических регрессионных моделей $f(\mathbf{w}, \mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle$ — параметрических функций, линейных относительно параметров. Функция ошибки задана следующим образом

$$S = \sum_{i \in \mathcal{X}} (y_i - f(\mathbf{w}, \mathbf{x}_i))^2, \tag{2}$$

где $\mathcal{X} \subseteq \mathcal{I}$ —некоторое множество индексов. Требуется найти такое подмножество индексов $\mathcal{A} \subseteq \mathcal{J}$, которое бы доставляло минимум функции

$$\mathcal{A}^* = \arg\min_{\mathcal{A} \subseteq \mathcal{J}} S(f_{\mathcal{A}} | \mathbf{w}^*, \mathcal{D}_{\mathcal{C}}) \tag{3}$$

на множестве индексов $\mathcal{C}$. При этом параметры $\mathbf{w}^*$ модели должны доставлять минимум функции

$$\mathbf{w}^* = \arg\min_{\mathbf{w} \in \mathcal{W}} S(\mathbf{w} | \mathcal{D}_{\mathcal{L}}, f_{\mathcal{A}}) \tag{4}$$

на множестве индексов $\mathcal{L}$. Здесь $f_{\mathcal{A}}$ обозначает модель f, включающую только столбцы матрицы X с индексами из множества $\mathcal{A}$, а обозначение вида $S(\mathbf{w}|\mathcal{D})$ означает, что переменная $\mathcal{D}$ фиксирована, а переменная $\mathbf{w}$ изменяется.

Выбор признаков при прогнозировании

Процедура выбора оптимального набора признаков. Опишем два этапа алгоритма: Add и Del. На первом этапе последовательно добавляются признаки, согласно (4), доставляющие минимум S на обучающей выборке, заданной множеством индексов $\mathcal{L}$. На втором этапе происходит последовательное удаление признаков, согласно методу Белсли. Пусть на k-ом шаге алгоритма имеется активный набор признаков $\mathcal{A}_k \in \mathcal{J}$. На нулевом шаге $\mathcal{A}_0$ пуст.

Этап Add. Находим признак доставляющий минимум S на обучающей выборке

$$j^* = \arg\min_{j \in \mathcal{J} \setminus \mathcal{A}_{k-1}} S(\mathbf{w} | \mathcal{D}_{\mathcal{L}}, f_{\mathcal{A}_{k-1} \cup \{j\}}).$$

Затем добавляем новый признак j^* к текущему активному набору

$$\mathcal{A}_k = \mathcal{A}_{k-1} \cup \{j^*\}$$

и повторяем эту процедуру до тех пор, пока $S(f_{\mathcal{A}_k} | \mathbf{w}^*, \mathcal{D})$ превосходит свое минимальное значение на данном этапе не более, чем на некоторое заданное значение ΔS_1.

Этап Del. Находим индексы обусловленности и долевые коэффициенты для текущего набора признаков $\mathcal{A}_{k-1}$ согласно методу Белсли, описание которого приведено ниже. Далее находим количество достаточно больших индексов обусловленности. Достаточно большими будем считать индексы квадрат которых превосходит максимальный индекс обусловленности η_t, где $t = |A_{k-1}|$ количество признаков в текущем наборе $\mathcal{A}_{k-1}$.

$$i^* = \sum_{g=1}^{t} \left[\eta_g^2 > \eta_t \right]. \tag{5}$$

Затем ищем в матрице долевых коэффициентов $\mathbf{var}(\mathbf{w})$ столбец j^* с максимальной суммой по последним i^* долевым коэффициентам

$$j^* = \arg\max_{j \in \mathcal{A}_{k-1}} \sum_{g=t-i^*+1}^{t} q_g^j. \tag{6}$$

Удаляем j^*-ый признак из текущего набора

$$\mathcal{A}_k = \mathcal{A}_{k-1} \backslash j^*$$

и повторяем эту процедуру до тех пор, пока $S(f_{\mathcal{A}_k}|\mathbf{w}^*, \mathcal{D})$ превосходит свое минимальное значение на данном этапе не более, чем на некоторое заданное значение ΔS_2.

Повторение этапов Add и Del осуществляется до тех пор, пока значение $S(f_{\mathcal{A}_k}|\mathbf{w}^*, \mathcal{D})$ не стабилизируется.

Метод Белсли для удаления признаков. Рассмотрим матрицу признаков $\mathbf{X}$. Она имеет размерность $m \times n$. Выполним ее сингулярное разложение:

$$\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T,$$

где $\mathbf{U}$, $\mathbf{V}$ — ортогональные матрицы размерностью соответственно $m \times m$ и $n \times n$ и $\mathbf{\Lambda}$ — диагональная матрица с элементами (сингулярными числами) на диагонали такими, что

$$\lambda_1 > \lambda_2 > \ldots > \lambda_r,$$

где r — ранг матрицы $\mathbf{X}$. Заметим, что в нашем случае $r = n$. Это связано с тем, что в алгоритме шагового выбора на каждом шаге мы имеем мультиколлиниарный, но невырожденный набор признаков. Столбцы матрицы $\mathbf{V}$ являются собственными векторами, а квадраты сингулярных чисел — собственными значениями матрицы $\mathbf{X}^T\mathbf{X}$.

$$\mathbf{X}^T\mathbf{X} = \mathbf{V}\mathbf{\Lambda}^T\mathbf{U}^T\mathbf{U}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{V}\mathbf{\Lambda}^2\mathbf{V}^T,$$
$$\mathbf{X}^T\mathbf{X}\mathbf{V} = \mathbf{V}\mathbf{\Lambda}^2.$$

Отношение максимального сингулярного числа к j-му сингулярному числу назовем индексом обусловленности с номером j

$$\eta_j = \frac{\lambda_{\max}}{\lambda_j}.$$

Если матрица X неполноранговая, то значительная часть индексов обусловленности неопределено. Однако, в нашем случае, как упомяналось выше, матрица признаков X является матрицей полного ранга.

Так как модель линейна, то $\mathbf{w} = \mathbf{B}\mathbf{y}$, где $\mathbf{w}$ — вектор параметров модели. То есть $w_i = \mathbf{b}_i^T\mathbf{y}$, где

$$\mathbf{B} = \begin{pmatrix} \mathbf{b}_1^T \\ \ldots \\ \mathbf{b}_n^T \end{pmatrix}.$$

Мы ищем несмещенную оценку параметров

$$\mathrm{E}(\mathbf{w}) = \mathbf{w} = \mathbf{B}\mathbf{X}\mathbf{w},$$

то есть $\mathbf{B}\mathbf{X} = \mathbf{I}$, где $\mathbf{I}$ — единичная матрица.

Тогда ковариация параметров w_i и w_j равна

$$cov(w_i, w_j) = \mathrm{E}(\mathbf{b}_i^T\mathbf{y} - \mathbf{b}_i^T\mathbf{X}\mathbf{w})(\mathbf{b}_j^T\mathbf{y} - \mathbf{b}_j^T\mathbf{X}\mathbf{w}) = \mathbf{b}_i^T\mathrm{E}((\mathbf{y} - \mathbf{X}\mathbf{w})(\mathbf{y} - \mathbf{X}\mathbf{w})^T))\mathbf{b}_j =$$

$$= \mathrm{E}(\xi_i\xi_j^T)\mathbf{b}_i^T\mathbf{b}_j = \sigma^2\mathbf{b}_i^T\mathbf{b}_j,$$

где ξ_i — i-ый регрессионный остаток, а σ^2 — дисперсия регрессионных остатков.

Мы хотим найти несмещенную оценку параметров, минимизирующую дисперсию параметров по каждой компоненте

$$\begin{cases} \sigma^2 \mathbf{b}_i^T \mathbf{b}_i \longrightarrow \min\limits_{\mathbf{B}} \\ \mathbf{b}_i^T \mathbf{X} = \mathbf{e}_i^T \end{cases},$$

где $\mathbf{e}_i^T$ — i-ая строка единичной матрицы. Составим фунцию Лагранжа

$$Ł = \mathbf{b}_i^T \mathbf{b}_i + \mathbf{\Lambda}_i^T (\mathbf{X}^T \mathbf{b}_i - \mathbf{e}_i),$$

где $\mathbf{\Lambda} = (\mathbf{\Lambda}_1 \dots \mathbf{\Lambda}_n)$. Продифференцировав по $\mathbf{b}_i$, получим

$$\begin{cases} 2\mathbf{b}_i + \mathbf{X}\mathbf{\Lambda}_i \\ \mathbf{X}^T \mathbf{b}_i - \mathbf{e}_i = 0 \end{cases}$$

Из первого уравнения $\mathbf{b}_i = -\frac{1}{2}\mathbf{X}\mathbf{\Lambda}_i$, тогда $-\frac{1}{2}\mathbf{X}^T\mathbf{X}\mathbf{\Lambda}_i = \mathbf{e}_i$. То есть $\mathbf{\Lambda} = -2(\mathbf{X}^T\mathbf{X})^{-1}$, и, окончательно, для $\mathbf{B}$ получим

$$\mathbf{B} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T.$$

Для ковариационной матрицы $\mathbf{A}$ получим

$$\mathbf{A} = \sigma^2 \mathbf{B}\mathbf{B}^T = \sigma^2 (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}((\mathbf{X}^T\mathbf{X})^{-1})^T = \sigma^2 \mathbf{X}^{-1}(\mathbf{X}^T)^{-1}\mathbf{X}^T\mathbf{X}((\mathbf{X}^T\mathbf{X})^{-1})^T =$$
$$= \sigma^2((\mathbf{X}^T\mathbf{X})^{-1})^T = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}.$$

В общем случае, выражение $\sigma^2(\mathbf{X}^T\mathbf{X})^{-1}$ является несмещенной оценкой ковариационной матрицы признаков, а в случае линейной модели оно в точности совпадает с ковариационной матрицей, то есть $\mathbf{A}^{-1} = \sigma^{-2}\mathbf{X}^T\mathbf{X}$.

Используя сингулярное разложение, дисперсия параметров, найденных методом наименьших квадратов $\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$, может быть записана как

$$\mathbf{var}(\mathbf{w}) = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1} = \sigma^2(\mathbf{V}^T)^{-1}\mathbf{\Lambda}^{-2}\mathbf{V}^{-1} = \sigma^2 V \mathbf{\Lambda}^{-2} V^T.$$

Таким образом, дисперсия j-го регрессионного коэффициента — это j-й диагональный элемент матрицы $\mathbf{var}(\mathbf{w})$.

Для обнаружения мультиколлинеарности признаков построим таблицу, в которой каждому индексу обусловленности η_j соответствуют значения q_{ij} — долевые коэффициенты. Сумма долевых коэффициентов по индексу j равна единице.

$$\sigma^{-2}\mathbf{var}(w_i) = \sum_{j=1}^{n} \frac{v_{ij}^2}{\lambda_j^2} = (q_{i1} + q_{i2} + \ldots + q_{in}) \sum_{j=1}^{n} \frac{v_{ij}^2}{\lambda_j^2},$$

где q_{ij} — отношение соответствующего слагаемого в разложении вектора $\sigma^{-2}\mathbf{var}(w_i)$ ко всей сумме, а $\mathbf{V} = (v_{ij})$.

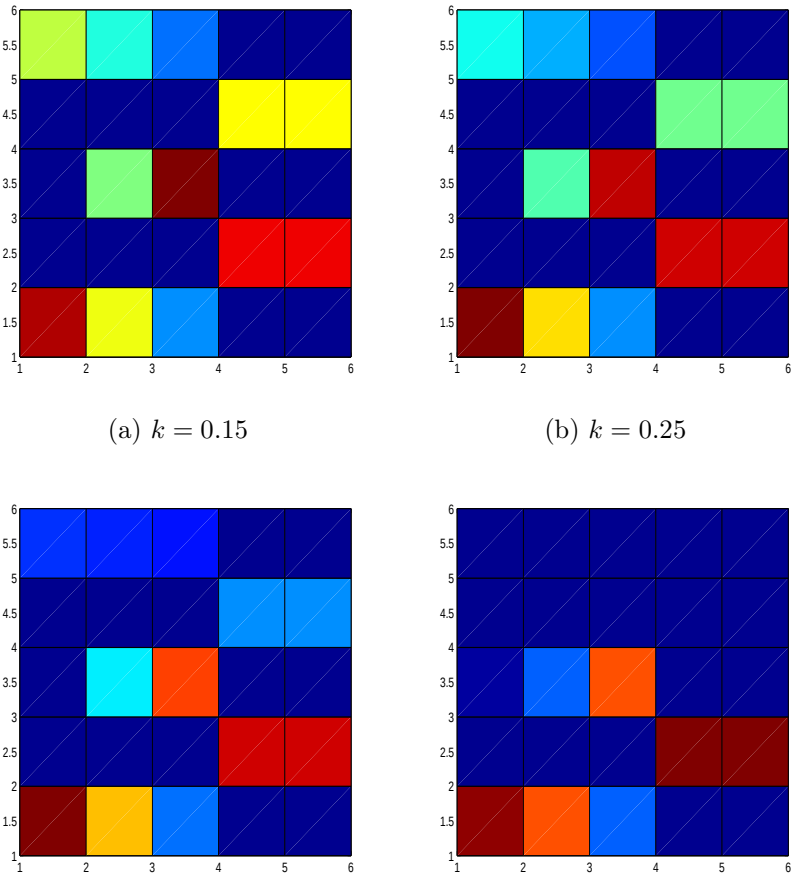
Таблица 1. Разложение $\mathbf{var}(\mathbf{w})$

Индекс обусловленности	$\mathbf{var}(w_1)$	$\mathbf{var}(w_2)$	$\dots$	$\mathbf{var}(w_n)$
η_1	q_{11}	q_{21}	$\dots$	q_{n1}
η_2	q_{12}	q_{22}	$\dots$	q_{n2}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
η_n	q_{1n}	q_{2n}	$\dots$	q_{nn}

Чем больше значение долевого коэффициента q_{ij} тем больший вклад вносит j-ый признак в дисперсию i-го регрессионного коэффициента.

Из таблицы (1) определяется мультиколлинеарность: большие величины η_j означают, что, возможно, есть зависимость между признаками. Если присутствует только один достаточно большой индекс обусловленности, тогда возможно определение участвующих в зависимости признаков из долевых коэффициентов: признак считается вовлеченным в зависимость, если его долевой коэффициент связанный с этим индексом превышает выбранный порог (обычно 0.25). Если же присутствует несколько больших индексов обусловленности, то вовлеченность признака в зависимость определяется по сумме его дисперсионных долей, отвечающих большим значениям индекса обусловленности: если сумма превышает выбранный порог, то признак участвует как минимум в одной линейной зависимости. Для нахождения мультиколлиниарных признаков решаются задачи (5) и (6).

Проиллюстрируем метод Белсли на примере. Используется неизменные признаки x_1, x_5 и зависящие от параметра k признаки x_2, x_3, x_4. При $k = 0$ все признаки ортогональны, при увеличении k признаки x_2, x_3 приближаются к x_1, а x_4 — к x_5 вплоть до полной коллинеарности при $k = 1$. На рис. 1 приведены матрицы долевых коэффициентов в зависимости от k.



(a) $k = 0.15$ (b) $k = 0.25$

(c) $k = 0.4$ (d) $k = 0.9$

Рис. 1. Матрицы долевых коэффициентов

В таблице (2) приведены значения индексов обусловленности в зависимости от k.

Таблица 2. Индексы обусловленности

k	**0.15**	**0.25**	**0.4**	**0.9**
	1.0	1.0	1.0	1.0
	1.0	1.0	1.1	1.2
	1.1	1.2	1.5	21.5
	1.2	1.4	2.0	22.1
	1.2	1.5	2.1	24.0

Наблюдается две основных зависимости — первая между признаками x_1, x_2, x_3 и вторая между признаками x_4, x_5.

Подсчет матожижания и дисперсии функции ошибки

Функцию ошибки S можно при фиксированном наборе признаков $\mathcal{A} \in \mathcal{J}$ считать случайной величиной. Мы хотим минимизировать ее математическое ожидание и дисперсию при фиксированной сложности модели.

Сначала проведем эмпирический анализ наших реальных данных, а затем сравним полученные результаты с статистическими.

Эмпирический подход. Для данного набора признаков $\mathcal{A} \in \mathcal{J}$ будем многократно разбивать выборку на обучение $\mathcal{L}$ и контроль $\mathcal{C}$. Полученные значения функции ошибки S можно считать реализациями случайной величины. Тогда математическое ожидание и дисперсия оцениваются следующим образом

$$ES = \frac{1}{m}\sum_{i=1}^{m} S_i,$$

$$DS = \frac{1}{m}\sum_{i=1}^{m} (S_i - ES)^2,$$

где m — число разбиений выборки, а S_i — значение функции ошибки при i-ом разбиении.

Ниже представлены графики полученные по данным прогрессирования заболевания у больных диабетом. На нем отмечены все 2^n точек, где $n = 10$ — число признаков. По вертикали отложена дисперсия в логарифмическом маштабе, а по горизонтали количество признаков в наборе. При каждом значении числа признаков (сложности модели) найден набор с минимальным математическим ожиданием финкции ошибки — эти точки отмечены красным.

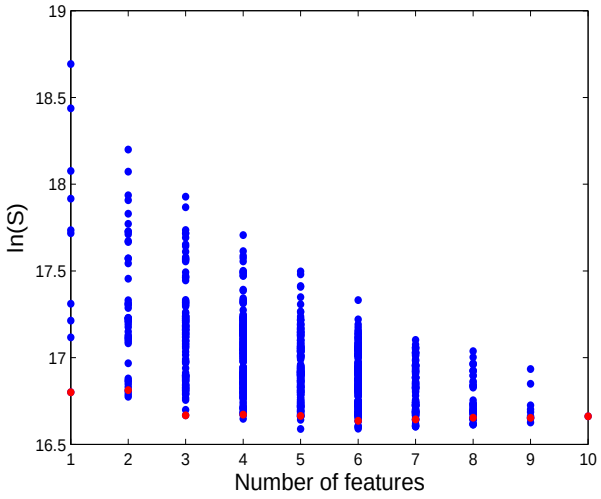


Рис. 2. Зависимость логарифма дисперсии функции ошибки от числа признаков в наборе при leave-one-out

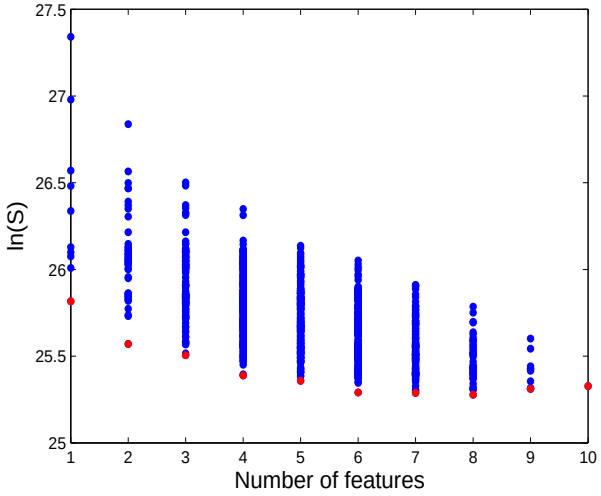


Рис. 3. Зависимость логарифма дисперсии функции ошибки от числа признаков в наборе при случайном разбиении выборки

По графикам видно, что у наборов с малым математическим ожиданием функции ошибки дисперсия тоже мала.

Статистический подход. Мы предполагаем, что данные нормальные, то есть

$$\mathbf{y} \sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I}),$$

где σ^2 — дисперсия регресиионных остатков, а $\mathbf{I}$ — единичная матрица.

При таких предположениях в методе наименьших квадратов функция ошибки $S(\mathbf{w}) =$ $=$ SSE имеет известное распределение

$$\frac{S}{\sigma^2} \sim \chi^2(m-n),$$

где m — число объектов (строк в матрице $\mathbf{X}$), а n — число признаков [15]. Из свойств распределения χ^2 получим

$$E\frac{S}{\sigma^2} = m - n,$$

$$D\frac{S}{\sigma^2} = 2(m - n).$$

То есть

$$ES = (m - n)\sigma^2,$$

$$DS = 2(m - n)\sigma^4,$$

причем σ^2 своя для каждого набора признаков.

Таким образом получаем, что математическое ожидание функции ошибки S достигает минимума при заданной сложности модели на том же наборе, на котором дисперсия достигает минимума. Этот результат экспериментально подтвердился на наших данных.

Путь в n-мерном кубе

В нашей задаче мы имеем дело с n признаками, то есть существует 2^n возможных наборов признаков, из которых мы пытаемся найти оптимальный. Все эти 2^n наборов можно представить как вершины n-мерного куба. В данной работе используется шаговый алгоритм поиска оптимального набора, то есть пошагового движения по вершинам этого куба.

Приведем пример движения по вершинам куба при работе предложенного алгоритма. Всего использовалось 6 признаков $x_1, \ldots, x_6$, они изображены на рис.4. Также на нем показан вектор ответов y.

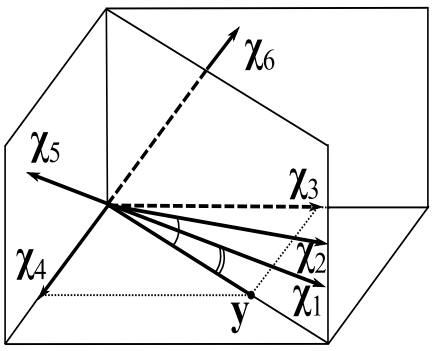


Рис. 4. Данные

На рис.5 показан путь по вершинам куба для описанных данных. По вертикали отложен номер признака, по горизонтали — номер итерации. Красная клетка означает, что признак на данной итерации вошел в набор, синяя — не вошел. Например признак номер 6 присутствовал в наборе с 3 по 8 итерацию, но в конечный набор не вошел.

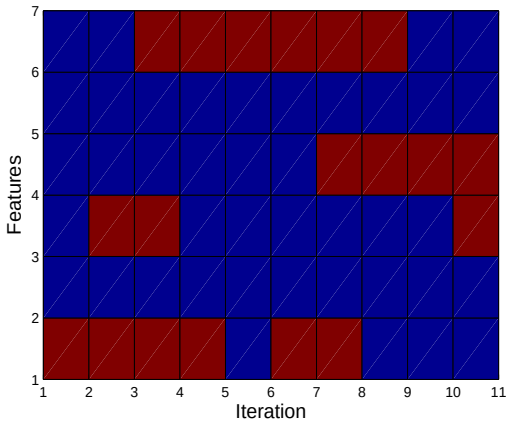


Рис. 5. Путь в кубе

Сформулируем и докажем некоторые теоретические утверждения, связанные с движением в кубе. Пошаговый алгоритм (1) выбора набора признаков.

Этап Add. Последовательно добавляются признаки, доставляющие минимум S

$$j^* = \arg \min_{j \in \mathcal{J} \setminus \mathcal{A}_{k-1}} S(\mathbf{w}|\mathcal{D}_{\mathcal{L}}, f_{\mathcal{A}_{k-1} \cup \{j\}}).$$

$$\mathcal{A}_k = \mathcal{A}_{k-1} \cup \{j^*\}$$

и повторяем эту процедуру до тех пор, пока $S(f_{\mathcal{A}_k}|\mathbf{w}^*, \mathcal{D})$ превосходит свое минимальное значение на данном этапе не более, чем на некоторое заданное значение ΔS_1.

Этап Del. Последовательно удаляем признаки, согласно методу Белсли

$$i^* = \sum_{g=1}^{t} \left[\eta_g^2 > \eta_t\right]$$

$$j^* = \arg \max_{j \in \mathcal{A}_{k-1}} \sum_{g=t-i^*+1}^{t} q_g^j$$

$$\mathcal{A}_k = \mathcal{A}_{k-1} \backslash j^*$$

и повторяем эту процедуру до тех пор, пока $S(f_{\mathcal{A}_k}|\mathbf{w}^*, \mathcal{D})$ превосходит свое минимальное значение на данном этапе не более, чем на некоторое заданное значение ΔS_2.

Повторение этапов Add и Del осуществляется до тех пор, пока значение $S(f_{\mathcal{A}_k}|\mathbf{w}^*, \mathcal{D})$ не стабилизируется.

Алгоритм (1) дает нам на выходе решение, которым является одна из вершин куба, то есть некоторый набор признаков.

Определение 1. Смежными будем называть вершины, одна из которых получается из другой добавлением одного признака.

Утверждение 1. Значение функции ошибки S в вершине-решении меньше, чем ее значение в любой смежной с ней вершине большей мощности.

Доказательство Предположим противное, и в одной из смежных вершин значение функции ошибки S меньше, чем в вершине-решении, тогда алгоритм (1) сделал бы еще один шаг добавления, но алгоритм остановился. Получили противоречие.

Определение 2. Путем в кубе будем называть последовательность вершин, соответствующую последовательности наборов признаков в пошаговом алгоритме (1). Первым членом последовательности всегда является пустой набор, а последним — решение полученное алгоритмом (1).

Определение 3. Сегментом пути назовем отрезок последовательности вершин, определенной данным путем в кубе.

Определение 4. Сегментом типа Add назовем сегмент, в котором все вершины получены добавлением нового признака.

Определение 5. Сегментом типа Del назовем сегмент, в котором все вершины получены удалением одного признака из набора.

Утверждение 2. Если путь в кубе имеет два одинаковых члена последовательности, принадлежащих некоторым сегментам одного типа, то решение полученное алгоритмом (1) равно одному из членов конечной последовательности, образованной уже пройденной частью пути.

Доказательство. Если мы попадаем на p-ом шаге в вершину в которой уже были на шаге t, причем на сегменте пути того же типа, то путь по кубу начиная с t-ого шага, совпадает с уже пройденным с p-го по t-ый шаг участком пути. Таким образом, решение, полученное алгоритмом (1), равно одному из членов конечной последовательности, образованной уже пройденной частью пути.

Утверждение 3. Значение функции ошибки S в вершине, являющейся решением, построенным с помощью алгоритма (1), меньше, чем ее значение в любой из вершин смежных с некоторой вершиной пути **a** и имеющей большее число признаков в наборе, чем вершина **a**.

Доказательство. Предположим противное, то есть значение функции ошибки S в вершине **b** смежной с вершиной пути **a**, принадлежащей сегменту типа Add меньше, чем в вершине-решении. Но тогда вершина **b** принадлежит пути, и алгоритм (1) выдаст ее в качестве решения. Получили противоречие.

Утверждение 4. Два различных пути, проходящие через одну вершину на сегментах пути одного типа, при дальнейшем движении по кубу совпадают.

Доказательство. Данное утверждение объясняется тем, что алгоритм (1) однозначно строит путь, выходящий из данной вершины и принадлежащий сегменту определенного типа.

Заключение

В работе предложен метод поиска оптимальной модели, основанный на комбинации двух стратегий: отбор признаков и выбор модели. Особенно полезен предложенный метод в случае, когда данные содержат большое число мультиколлинеарных признаков. Предложенный алгоритм позволяет получать хорошо обусловленные наборы порожденных признаков. В работе теоретически обоснованно, что математическое ожидание функции ошибки S достигает минимума при заданной сложности модели на том же наборе, на котором дисперсия достигает минимума. Этот результат так же подтвержден экспериментально на реальных данных.

Литература

[1] Frisch R. *Statistical Confluence Analysis by means of complete regression systems*, Universitetets Okonomiske Institute, 1934.

[2] Efron B., Hastie T., Johnstone I., Tibshirani R. *Least angle regression*, The Annals of Statistics, 2004, Vol. 32, no. 3., Pp. 407-499.

[3] Tibshirani R. *Regression shrinkage and selection via the lasso*, Journal of the Royal Statistical Society, 1996, Vol. 32, no. 1, Pp. 267-288.

[4] Draper N. R., Smith H. *Appied Regression Analysis*,John Wihley and Sons, 1998.

[5] Chen Y. W., Billings C. A., Luo W. *Orthogonal least squares methods and their application to non-linear system identication*, International Journal of Control, 1989, Vol. 2, no. 50, Pp. 873-896.

[6] Chen S., Cowan C. F. N., Grant P. M. *Orthogonal least squares learning algorithm for radial basis function network*, Transaction on neural netwark, 1991, Vol. 2, no. 2, Pp. 302-309.

[7] Efroymson M. A. *Multiple regression analysis*, New York: Ralston, Wiley, 1960.

[8] Rawlings J. O., Pantula S. G., Dickey D. A. *Appied Regression Analysis: A Research Tool*, New York: Springer-Verlag, 1998.

[9] Belsley D. A. *Conditioning Diagnostics: Collinearitu and Weak Data in Regression*, New York: John Wiley and Sons, 1991.

[10] Tarantola A. *Inverse Problem Theory*, SIAM, 2005.

[11] Johnstone I., Tibshirani R., Efron B., Hastie T. *Least Angle Regression*, 2004.

[12] McNames J. *Innovations in Local Modeling for Time Series Prediction*, 1999.

[13] Федорова В. П. *Локальные Методы Прогнозирования Временных Рядов*, 2009.

[14] Е. А. Крымова, В. В. Стрижов *Выбор моделей в линейном регрессионном анализе*, Информационные технологии, 2011.

[15] Г. И. Ивченко, Ю. И. Медведев *Введение в математическую статитсику*, ЛКИ, 2009.

Оценка гиперпараметров линейных регрессионных моделей методом максимального правдоподобия при отборе шумовых и коррелирующих признаков*

А. А. Зайцев[1,2], *А. А. Токмакова*[1]
likzet@gmail.com, aleksandra-tok@yandex.ru
1 — Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»
2 — Datadvance

Рассматривается задача выбора регрессионной модели. Предполагается, что вектор параметров модели — многомерная случайная величина с независимо распределёнными компонентами. В работе предложен способ оптимизации параметров и гиперпараметров. Приведены явные оценки гиперпараметров для случая линейных и нелинейных моделей. Показано как полученные оценки используются для отбора признаков. Предложенный подход сравнивается с подходом, использующим для оценки гиперпараметров аппроксимацию Лапласа.

Ключевые слова: *регрессия, выбор признаков, распределение параметров, оценка гиперпараметров, байесовский вывод.*

Введение

В данной работе рассматривается задача выбора регрессионной модели [6] из заданного параметрического семейства регрессионных моделей. Один из возможных подходов — введение предположения о распределении параметров модели [8]. В этом случае предполагается, что функция регрессии задана оценкой вектора параметров, который считается нормально распределенно многомерной случайной величиной. Параметры распределения заданы вектором, в дальнейшем называемым вектором гиперпараметров модели.

Впервые этот подход к выбору признаков методом анализа распределения параметров был предложен в работе [7]. Более общий подход был предложен Маккаем в работе [8]. В этой работе Маккай ввел понятие гиперпараметров. Бишоп предложил ряд других способов оценки гиперпараметов, таких как Марковские цепи Монте-Карло и аппроксимация Лапласа [5, 4]. Подход, использующий аппроксимацию Лапласа был развит в работах [3, 2].

Предлагается для линейной регрессионной модели выписать явное выражение функции правдоподобия с учётом введенных вероятностных предположений. Максимизируя правдоподобие, получаем оценки наиболее правдоподобных значений гиперпараметров модели. Такой подход позволяет получать оценки гиперпараметров регрессионных моделей. Для полученных оценок гиперпараметров явно выписываются оценки параметров модели. Они используются для отбора признаков. Предложенный подход сравнивается с подходом, использующим аппроксимацию Лапласа распределения параметров модели [3].

Во второй части работы дана постановка задачи и принятая гипотеза порождения данных. В третей части получена оценка правдоподобия гиперпараметров линейной модели и описан подход, позволяющий оценивать гиперпараметры, доставляющие максимум правдоподобия. В четвертой части описан подход, позволяющий оценивать гиперпараметры, максимизирующие правдоподобие для нелинейных регрессионных моделей с использованием аппроксимации Лапласа. В пятой части описан процедура отбора признаков, ис-

Научный руководитель В. В. Стрижов

пользующая полученные значения гиперпараметров. В шестой части проведено сравнение предложенного и используемых подходов на модельных и реальных данных.

Постановка задачи

Задана выборка $D = (X, \mathbf{y}) = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$, где $\mathbf{x}_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$. Рассматривается класс регрессионных моделей вида:

$$\mathbf{y} = \mathbf{f}(X, \mathbf{w}) + \boldsymbol{\varepsilon}. \tag{1}$$

Предполагается, что шум $\boldsymbol{\varepsilon}$ — многомерная нормальная случайная величина с нулевым математическим ожиданием и матрицей ковариации B^{-1}:

$$\boldsymbol{\varepsilon} \sim \mathcal{N}(0, B^{-1}), \tag{2}$$

вектор параметров модели $\mathbf{w}$ — многомерная нормальная случайная величина с нулевым математическим ожиданием и матрицей ковариации A^{-1}:

$$\mathbf{w} \sim \mathcal{N}(0, A^{-1}). \tag{3}$$

Требуется получить оценки матриц A, B согласно гипотезе порождения данных (2), (3).

Правдоподобие для линейной модели

Рассмотрим линейную регрессионную модель. Тогда (1) имеет вид

$$\mathbf{y} = X\mathbf{w} + \boldsymbol{\varepsilon}.$$

Плотность распределения параметров $\mathbf{w}$ согласно теории Байеса имеет вид:

$$p(\mathbf{w}|A, B, D, f) = \frac{p(D|\mathbf{w}, B)p(\mathbf{w}|A)}{p(D|A, B)}, \tag{4}$$

в котором $p(D|\mathbf{w}, B)$, $p(\mathbf{w}|A)$ — плотности многомерных нормальных случайных величин:

$$p(D|\mathbf{w}, B) = \frac{1}{(2\pi)^{\frac{m}{2}}|B|^{-\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{y} - X\mathbf{w})^T B(\mathbf{y} - X\mathbf{w})\right), \tag{5}$$

согласно предположению о нормальности распределения шумов (2), и

$$p(\mathbf{w}|A) = \frac{1}{(2\pi)^{\frac{n}{2}}|A|^{-\frac{1}{2}}} \exp\left(-\frac{1}{2}\mathbf{w}^T A\mathbf{w}\right), \tag{6}$$

согласно предположению о распределении вектора параметров модели (3). Правдоподобие модели $p(D|A, B)$ имеет вид

$$p(D|A, B) = \int_{\mathbb{R}^n} p(D|\mathbf{w}, A, B)p(\mathbf{w}|A, B)d\mathbf{w}. \tag{7}$$

Для линейных моделей явно выпишем оценки гиперпараметров модели $p(D|A, B)$. Отметим, что в работе [3] был предложен подход, который оценивал эту вероятность с использованием аппроксимации Лапласа. Верна следующая теорема.

Теорема 1. *Правдоподобие в предположениях о распределении шума* $\boldsymbol{\varepsilon}$ (2) *и параметров модели* $\mathbf{w}$ (3) *имеет вид*

$$p(D|A,B) = \frac{|B|^{\frac{1}{2}}|A|^{\frac{1}{2}}}{(2\pi)^{\frac{m}{2}}|K|^{\frac{1}{2}}} \exp\left(\frac{1}{2}\mathbf{y}^T(C^TKC - B)\mathbf{y}\right), \tag{8}$$

а его логарифм имеет вид

$$\ln p(D|A,B) = -\frac{1}{2}\left(\ln|K| + m\ln 2\pi - \ln|B| - \ln|A| - \mathbf{y}^T(C^TKC - B)\mathbf{y}\right). \tag{9}$$

Здесь

$$K = X^TBX + A, \quad C = K^{-1}X^TB.$$

Доказательство.

Подставляя (5) и (6) в (7) получим следующее выражение:

$$p(D|A,B) =$$

$$= \int_{\mathbb{R}^n} \frac{1}{(2\pi)^{\frac{m}{2}}|B|^{-\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{y} - X\mathbf{w})^TB(\mathbf{y} - X\mathbf{w})\right) \frac{1}{(2\pi)^{\frac{n}{2}}|A|^{-\frac{1}{2}}} \exp\left(-\frac{1}{2}\mathbf{w}^TA\mathbf{w}\right) d\mathbf{w} =$$

перепишем произведение двух экспонент как экспоненту от их суммы:

$$= \int_{\mathbb{R}^n} \frac{|B|^{\frac{1}{2}}|A|^{\frac{1}{2}}}{(2\pi)^{\frac{n+m}{2}}} \exp\left(-\frac{1}{2}\left((\mathbf{y} - X\mathbf{w})^TB(\mathbf{y} - X\mathbf{w}) + \mathbf{w}^TA\mathbf{w}\right)\right) d\mathbf{w} =$$

введем обозначения $K = A + X^TBX, C = K^{-1}X^TB$ и выделим

полный квадрат по $(\mathbf{w} - C\mathbf{y})$:

$$= \int_{\mathbb{R}^n} \frac{|B|^{\frac{1}{2}}|A|^{\frac{1}{2}}}{(2\pi)^{\frac{n+m}{2}}} \exp\left(-\frac{1}{2}\left((\mathbf{w} - C\mathbf{y})^TK(\mathbf{w} - Cy) - \mathbf{y}^T(C^TKC - B)\mathbf{y}\right)\right) d\mathbf{w} =$$

интеграл по плотности многомерного нормального распределения равен единице:

$$= \frac{|B|^{\frac{1}{2}}|A|^{\frac{1}{2}}}{(2\pi)^{\frac{m}{2}}|K|^{\frac{1}{2}}} \exp\left(\frac{1}{2}\left(\mathbf{y}^T(C^TKC - B)\mathbf{y}\right)\right).$$

Следовательно, искомое правдоподобие модели $p(D|A,B)$ имеет вид (8), а его логарифм — вид (9). ■

Рассмотрим теперь случай, когда матрица A — диагональная, а матрица $B = \beta I$.

Следствие 1. *Если матрица* A *— диагональная, а матрица* B *имеет вид* $B = \beta I$*, то логарифм правдоподобия модели* $\ln p(D|A,B)$ *имеет вид*

$$\ln p(D|A,\beta) = -\frac{1}{2}\left(\ln|K| + m\ln 2\pi - m\ln\beta - \ln|A| - \beta\mathbf{y}^T(\beta XK^{-1}X^T - I)\mathbf{y}\right),$$

где $K = A + \beta X^TX$.

Вычисление производных функции правдоподобия модели $\ln p(D|A,B)$ по гиперпараметрам A, B

Для поиска максимума правдоподобия будем пользоваться градиентными методами оптимизации [1], поэтому нам понадобится выражения для производных $\ln p(D|A,B)$ по гиперпараметрам A, B.

Пусть матрица A имеет вид $A = \{\alpha_{ij}\}, i, j = \overline{1,n}$, а матрица B имеет вид $B = \{\beta_{ij}\}, i, j = \overline{1,m}$. Обе матрицы являются симметричными и неотрицательно определенными, так как являются матрицами ковариации.

Верны следующие два свойства производных матриц [9]. Для симметричной матрицы M верно, что

$$\frac{\partial \ln |M|}{\partial t} = \operatorname{tr}\left(M^{-1}\frac{\partial M}{\partial t}\right),$$

где t — некоторый параметр, $M = M(t)$. Так же верно, что

$$\frac{\partial M^{-1}}{\partial t} = -M^{-1}\frac{\partial M}{\partial t}M^{-1}.$$

Введем обозначение S^{ij} — такая матрица, что для двух индексов k, l выполнено, что

$$S^{ij}_{kl} = \begin{cases} 1, & k = i, l = j \text{ или } k = j, l = i, \\ 0, & \text{иначе.} \end{cases}$$

Запишем производную $\ln p(D|A,\beta)$ по β_{ij}:

$$\frac{\partial \ln p(D|A,B)}{\partial \beta_{ij}} = -\frac{1}{2}\Big(\operatorname{tr}\left(K^{-1}X^T S^{ij} X\right) - \operatorname{tr}\left(B^{-1}S^{ij}\right) - \mathbf{y}^T\big(S^{ji}XK^{-1}X^TB + B^TXK^{-1}X^TS^{ij} - B^TXK^{-1}X^TS^{ij}XK^{-T}X^TB - S^{ij}\big)\mathbf{y}\Big).$$

Аналогично запишем производную $\ln p(D|A,\beta)$ по α_{ij}:

$$\frac{\partial \ln p(D|A,B)}{\partial \alpha_{ij}} = -\frac{1}{2}\Big(\operatorname{tr}\left(K^{-1}S^{ij}\right) - \operatorname{tr}\left(A^{-1}S^{ij}\right) + \mathbf{y}^TB^TXK^{-1}S^{ij}K^{-T}X^TB\mathbf{y}\Big).$$

Так же запишем производные в предположениях следствия 1, $A = \operatorname{diag}(\alpha_1, \alpha_2, \ldots, \alpha_n)$, $B = \frac{1}{\beta}I$.

$$\frac{\partial \ln p(D|A,\beta)}{\partial \beta} = -\frac{1}{2}\Big(\operatorname{tr}\left(K^{-1}X^TX\right) - \frac{m}{\beta} + \mathbf{y}^T(2\beta XK^{-1}X^T - I - \beta^2XK^{-1}X^TXK^{-1}X^T)\mathbf{y}\Big)$$

$$\frac{\partial \ln p(D|A,\beta)}{\partial \alpha_i} = -\frac{1}{2}\left(\operatorname{tr}\left(K^{-1}I^{ii}\right) - \frac{1}{\alpha_i} - \beta^2\mathbf{y}^TXK^{-1}I^{ii}K^{-1}X^T\mathbf{y}\right).$$

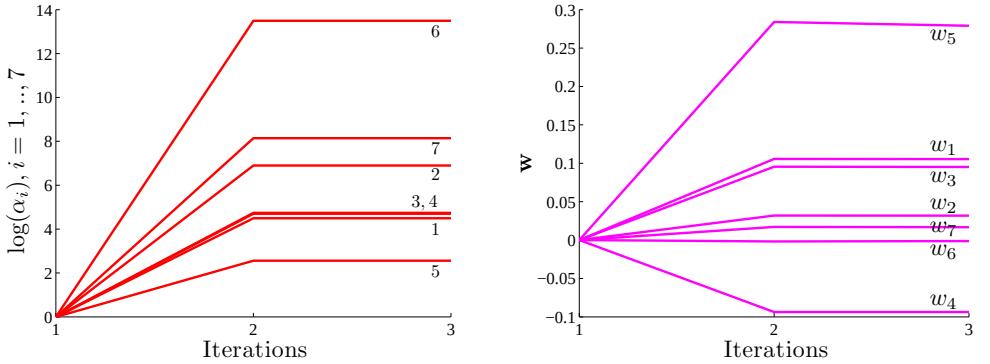
Так как получены значения производных правдоподобия модели $\ln p(D|A,B)$ по гиперпараметрам A, B, можно использовать любой градиентный метод оптимизации для поиска гиперпараметров A, B, максимизирующих правдоподобие модели.

Полученные значения гиперпараметров α_i, $i = 1, \ldots, n$ для диагональной матрицы A могут быть использованы для отбора признаков и выбора модели линейной регрессии.

Параметры w_i модели f сравниваются, используя оценки значений гиперпараметров α_i. Большие значения гиперпараметра α_i означают большой штраф на значение параметра и, следовательно, меньшую значимость данных параметров для качества модели. Малые значения α_i показывают большую значимость данного компонента модели для ее качества.

Модифицированный алгоритм Левенберга-Марквардта

Для минимизации функции ошибки воспользуемся алгоритмом Левенберга-Марквардта, который предназначен для оптимизации параметров нелинейных регрессионных моделей. Алгоритм заключается в последовательном приближении заданных начальных значений параметров к искомому локальному оптимуму и является обобщением метода сопряжённых градиентов и алгоритма Ньютона-Гаусса.

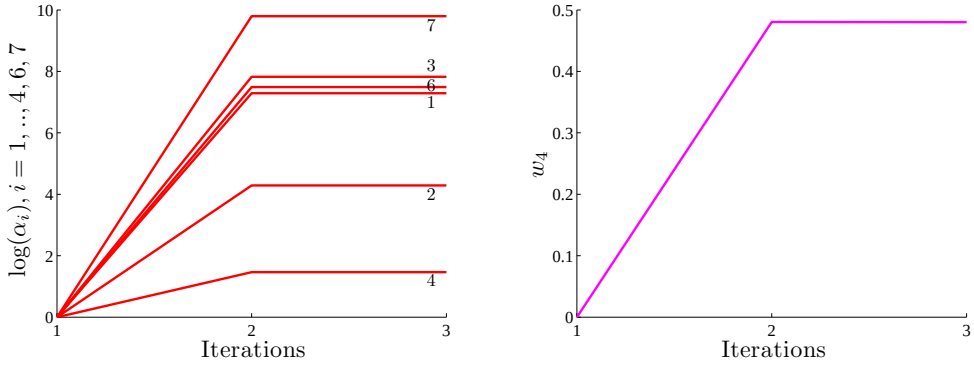


(a) Логарифм элементов матрицы A (b) Логарифм элементов матрицы A

Рис. 1. Исследование прочности при сжатии бетона

Пусть задано некоторое приближение для значений параметров модели $\mathbf{w}$. Тогда функция ошибки имеет вид:

$$S = \frac{1}{2}(\mathbf{w}+\Delta\mathbf{w})^T A(\mathbf{w}+\Delta\mathbf{w}) + \frac{1}{2}\big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big)^T B\big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big). \qquad (10)$$



(a) Логарифм элементов матрицы A (b) Параметр модели w_4

Рис. 2. Исследование морозостойкости бетона

На нулевой итерации алгоритма задаётся начальное приближение для $\mathbf{w}$. Приращение $\Delta\mathbf{w}$ в точке оптимума для функции ошибки (10) равно нулю. Поэтому для нахождения экстремума приравняем вектор частных производных S по $\mathbf{w}$ к нулю. Для этого представим S в виде двух слагаемых:

$$S_1 = \frac{1}{2}(\mathbf{w}+\Delta\mathbf{w})^T A(\mathbf{w}+\Delta\mathbf{w}),\ S_2 = \frac{1}{2}\big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big)^T B\big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big).$$

После дифференцирования получим следующие выражения:

$$\frac{\partial S_1}{\partial \mathbf{w}} = \frac{1}{2}(\mathbf{w}+\Delta\mathbf{w})^T(A+A^T),$$
$$\frac{\partial S_2}{\partial \mathbf{w}} = \frac{1}{2}\big[\big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big)^T B^T X + \big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big)^T BX\big].$$

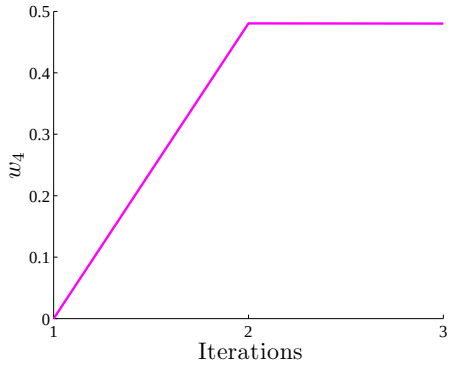


Рис. 3. Параметры модели $\mathbf{w}$

Таким образом, чтобы найти приращение $\Delta\mathbf{w}$ необходимо решить систему линейных уравнений:

$$\nabla S = \frac{1}{2}(\mathbf{w}+\Delta\mathbf{w})^T(A+A^T) + \frac{1}{2}\big[\big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big)^T B^T X + \big(X(\mathbf{w}+\Delta\mathbf{w})-\mathbf{y}\big)^T BX\big] = 0.$$

Выразив приращение $\Delta\mathbf{w}$, учитывая, что A, B симметричные матрицы, получим следующую рекуррентную формулу:

$$\Delta\mathbf{w} = \big[\big(A+X^T BX\big)^{-1}\big]^T\big(-\mathbf{w}^T A + (\mathbf{y}-\mathbf{X}\mathbf{w})^T BX\big)^T.$$

Алгоритм останавливается, в том случае, если приращение $\Delta\mathbf{w}$ в последующей итерации меньше заданного значения, либо если параметры $\mathbf{w}$ доставляют ошибку S меньшую заданной величины. Значение вектора $\mathbf{w}$ на последней итерации считается искомым.

Таблица 1. Численные значения параметров модели

w_1	w_2	w_3	w_4	w_5	w_6	w_7
0.1054	0.0317	0.0951	−0.0937	0.2790	−0.0013	0.0168

Вычислительный эксперимент

Результатом вычислительного эксперимента является фильтрация шумовых и коррелирующих признаков. Тестирование алгоритма производится на временном ряде, содержащем информацию о семи компонентах, входящих в состав бетона. Исследуется два отклика: предел прочности при сжатии и морозостойкость. Ряд содержит 103 записи. Необходимо построить регрессионную модель и оценить её параметры.

При исследовании предела прочности при сжатии алгоритм приводит к следующим результатам. На рис. **??** представлены логарифмы диагональных элементов матрицы A. Шестой элемент почти в два раза больше всех остальных, поэтому соответствующий ему параметр модели w_6 мал, как мы видим из графика **??** и 1. Однако α_6 не настолько велик, чтобы мы могли убрать соответствующий столбец матрицы плана, так как при этом произойдёт увеличение функции ошибки на 20%.

Таблица 2. Численные значения параметров модели

w_1	w_2	w_3	w_4	w_5	w_6	w_7
-0.0262	-0.1176	-0.0201	0.4801	0	-0.0238	-0.0079

При исследовании морозостойкости наблюдается вырождение матрицы A. Так на рисунке **??** приведён итерационный процесс для всех диагональных элементов α, кроме пятого, так как на третьей итерации $\log(\alpha_5)$ достигает значения 66, что в шесть раз превышает все остальные логарифмы элементов матрицы A. Рассматривая соответствующие графики **??**, 3 и таблицу 2, получим, что пятый признак является неинформативным и может быть исключен из матрицы плана. Функция ошибки увеличится менее, чем на 1%.

В обоих случаях использование аппроксимации Лапласа для вычисления правдоподобия приводит к увеличению функции ошибки менее, чем на 1%.

Выводы

В работе получено точное выражение для правдоподобия $\ln p(D|A, B)$ и предложен подход к его оптимизации. Так же проведено сравнение предложенного подхода с аппроксимацией Лапласа искомого правдоподобия. Использование точного выражения для вычисления правдоподобия позволяет получить наиболее точные оценки гиперпараметров.

Литература

[1] Ю. Е. Нестеров. *Введение в выпуклую оптимизацию.* МЦНМО, 2010.

[2] В. В. Стрижов. Поиск параметрической регрессионной модели в индуктивно заданном множестве. *Журнал вычислительных технологий*, 1:93–102, 2007.

[3] В. В. Стрижов and Р. А. Сологуб. Индуктивное порождение регрессионных моделей предполагаемой волатильности для опционных торгов. *Вычислительные технологии*, 14(5):102–113, 2009.

[4] C. Bishop. *Pattern Recognition And Machine Learning.* Springer, 2006.

[5] C. M. Bishop and M. E. Tipping. Bayesian regression and classification. In *Suykens, J., Horvath, G. et al., eds. Advances in Learning Theory: Methods, Models and Applications*, volume 190, pages 267–285. IOS Press, NATO Science Series III: Computer and Systems Sciences, 2000.

[6] K.P. Burnham and D.R. Anderson. *Model selection and multimodel inference: a practical information-theoretic approach.* Springer Verlag, 2002.

[7] Y. LeCun, J. Denker, S. Solla, R. E. Howard, and L. D. Jackel. Optimal brain damage. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems II*, San Mateo, CA, 1990. Morgan Kauffman.

[8] David J.C. MacKay. Choice of basis for laplace approximation. Technical report, Machine Learning, 1998.

[9] C.E. Rasmussen. Gaussian processes in machine learning. *Advanced Lectures on Machine Learning*, 1:63–71, 2004.

Оценка необходимого объема выборки пациентов при прогнозировании сердечно-сосудистых заболеваний*

А. П. Мотренко
pastt.petrovna@gmail.com
Московский физико-технический институт, ФУПМ, каф. "Интеллектуальные системы"

В работе описан алгоритм классификации пациентов, перенесших инфаркт и имеющих предрасположенность к инфаркту. Признаками для определения состояния пациента служат измерения концентрации белков в крови. Решается задача оценки параметров функции регрессии и выбора признаков в логистической регрессии. Предполагается, что объем данных недостаточен, поэтому в работе предлагается способ оценки необходимого объема выборки.

Ключевые слова: логистическая регрессия, выбор признаков, оценка объема выборки, прогноз предрасположенности к инфаркту.

Введение

Решается задача логистической регрессии [1], в основе которой лежит предположение о биномиальном распределении независимой переменной, и оцениваются параметры функции регрессии [2, 3].

Предполагается, что число измеряемых признаков избыточно. Требуется отыскать оптимальный набор признаков, эффективно разделяющий классы. Признаки в логистической регрессии как правило выбираются с помощью шаговой регрессии [4, 5]. В данной работе используется полный перебор, так как он дает экспертам гарантию, что рассмотрены все возможные сочетания признаков при выборе модели. При этом экспертами вводились ограничения на сложность модели. Задача выбора признаков поставлена с использованием площади под ROC-кривой [6] в качестве внешней функции ошибки.

Задача классификации сопряжена с оценкой минимального объема выборки, достаточного для проведения классификации. Для этого используются следующие методы:

1. Метод доверительных интервалов, в основе которого лежат статистические методы [7].
2. Метод оценки объема выборки в логистической регрессии, предложенный Демиденко [8, 9]. Этот способ также основан на на методах математической статистики, но в отличие от метода доверительных интервалов, учитывает распределение зависимой переменной и постановку задачи.
3. Метод скользящего контроля, позволяющий оценить необходимый объем выборки с точки зрения контроля над переобучением [10].
4. Сравнение плотностей распределения параметров модели на различных подвыборках с помощью расстояния Кулльбака-Лейблера [12].

При проведении вычислительного эксперимента были использованы данные [13], подготовленные специалистами парижской лаборатории анализа крови «Иммуноклин».

Задача классификации и оценка параметров

Дана выборка $D = \{(\mathbf{x}_i, y_i)\}, i = 1, \ldots, m$, состоящая из m объектов (пациентов), каждый из которых описывается n признаками (биомаркерами) $\mathbf{x}_i \in \mathbb{R}^n$ и принадлежит одному из двух классов $y_i \in \{0, 1\}$. Рассмотрим задачу логистической регрессии. Предполага-

Научный руководитель В. В. Стрижов

ется, что вектор ответов $\mathbf{y} = [y_1, \ldots, y_m]^T$ — бернуллиевский случайный вектор с независимыми компонентами $y_i \sim \mathcal{B}(\theta_i)$ и плотностью

$$p(\mathbf{y}|\mathbf{w}) = \prod_{i=1}^{m} \theta_i^{y_i}(1-\theta_i)^{1-y_i}. \tag{1}$$

Определим функцию ошибки следующим образом:

$$E(\mathbf{w}) = -\ln p(\mathbf{y}|\mathbf{w}) = -\sum_{i=1}^{m} y_i \ln \theta_i + (1-y_i)\ln(1-\theta_i). \tag{2}$$

Другими словами, функция ошибки есть логарифм плотности, или функции правдоподобия, со знаком минус. Требуется оценить вектор параметров $\hat{\mathbf{w}}$, доставляющий минумум функции ошибки:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^n} E(\mathbf{w}). \tag{3}$$

Вероятность принадлежности объекта к одному из двух классов определим как

$$f(\mathbf{x}_i^T\mathbf{w}) = \frac{1}{1+\exp(-\mathbf{x}_i^T\mathbf{w})} = \theta_i. \tag{4}$$

Для оценки параметров, воспользовавшись тождеством

$$\frac{df(\xi)}{d\xi} = f(1-f),$$

вычислим градиент функции $E(\mathbf{w})$:

$$\nabla E(\mathbf{w}) = -\sum_{i=1}^{m}\big(y_i(1-\theta_i) - (1-y_i)\theta_i\big)\mathbf{x}_i = \sum_{i=1}^{m}(\theta_i - y_i)\mathbf{x}_i = \mathbf{X}^T(\boldsymbol{\theta}-\mathbf{y}),$$

где вектор $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_m]^T$ и матрица $\mathbf{X} = \big[\mathbf{x}_1^T, \ldots, \mathbf{x}_m^T\big]^T$ состоит из векторов-описаний объектов.

Параметров оцениваются методом Ньютона-Рафсона. Введем обозначение $\boldsymbol{\Sigma}$ — диагональная матрица с элементами $\Sigma_{ii} = \theta_i(1-\theta_i)$, $i = 1, \ldots, m$. В качестве начального приближения $\mathbf{w} = [w_1, \ldots, w_n]^T$ вектора $\hat{\mathbf{w}}$ возьмем

$$w_j = \sum_{i=1}^{m} y_i(1-y_i), \quad j = 1, \ldots, n.$$

Оценка параметров $\mathbf{w}_{k+1}$ логистической регрессии (4) на $k+1$-м шаге итеративного приближения имеет вид

$$\mathbf{w}_{k+1} = \mathbf{w}_k - (\mathbf{X}^T\boldsymbol{\Sigma}\mathbf{X})^{-1}\mathbf{X}^T(\boldsymbol{\theta}-\mathbf{y}) = (\mathbf{X}^T\boldsymbol{\Sigma}\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{\Sigma}(\mathbf{X}\mathbf{w}_k - \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}-\mathbf{y})). \tag{5}$$

Процедура оценки параметров повторяется, пока евклидова норма разности $\|\mathbf{w}_{k+1} - \mathbf{w}_k\|_2$ не станет достаточно мала.

Алгоритм классификации имеет вид:

$$a(\mathbf{x}) = \operatorname{sign}\big(f(\mathbf{x}, \mathbf{w}) - c_0\big), \tag{6}$$

где c_0 — задаваемое в (7) пороговое значение (англ. cut-off) функции регрессии (4).

Вычисления качества прогноза. В данной работе для оценки качества прогноза и для выбора признаков используется площадь AUC под кривой ROC, то есть кривой в осях (FPR(ξ), TPR(ξ)), где

$$\mathrm{TPR} = \frac{1}{m}\sum_{i=1}^{m}[a(\mathbf{x}_i)=1][y_i=1]$$

есть доля объектов выборки, правильно классифицированных в пользу данного класса, и

$$\mathrm{FPR} = \frac{1}{m}\sum_{i=1}^{m}[a(\mathbf{x}_i)=1][y_i=0]$$

есть доля ошибочно классифицированных в пользу данного класса объектов выборки. Здесь используется обозначение индикаторной функции:

$$[y=1] = \begin{cases} 1, & y=1; \\ 0, & y\neq 1. \end{cases}$$

Таким образом, алгоритм тем лучше разделяет классы чем больше значение AUC.

Отыскание параметра c_0 алгоритма классификации. Каждая точка кривой ROC соответствует некоторому значению c_0. В алгоритме (6) используется то значение c_0, которое соответствует наибольшему расстоянию от отрезка [(0,0);(1,1)], означающего отказ от принятия решения о классификации, до кривой ROC:

$$\hat{\sigma}_0 = \arg\max_{\xi\in[0,1]} \left\| \big(\mathrm{TPR}(\xi), \mathrm{FPR}(\xi)\big) - (\xi,\xi) \right\|_1 . \tag{7}$$

Последнее выражение включает вычисление значения функционала качества, и как следствие, вычисление выражения (6) и итеративную оценку параметров (5).

Выбор признаков в задаче классификации

Введем обозначения $\mathcal{A}$ — некоторое подмножество индексов признаков, $\mathcal{A} \subseteq \mathcal{J} = \{1,\ldots,n\}$ и $\hat{\mathcal{A}}$ — оптимальный набор индексов. Обозначим $\mathbf{X}_\mathcal{A}$ множество столбцов-признаков матрицы $\mathbf{X}$, заданное набором $\mathcal{A}$, и $\mathbf{w}_\mathcal{A}$ — соответствующие им параметры. Рассмотрим задачу выбора признаков как задачу максимизации:

$$\hat{\mathcal{A}} = \arg\max_{\mathcal{A}\subseteq\mathcal{I}} S(\mathcal{A}) \text{ при условии } |\mathcal{A}| = \mathrm{const}. \tag{8}$$

В задаче использована площадь под кривой $S(\mathcal{A}) \equiv S(\mathbf{X}_\mathcal{A}, \hat{\mathbf{w}}_\mathcal{A}, \hat{\sigma}_0, \mathbf{y})$, значение которой вычислено для набора индексов признаков $\mathcal{A}$, а параметры $\hat{\mathbf{w}}_\mathcal{A}$ и c_0 получены в результате решения задач (3) и (7).

Набор признаков отыскивается путем полного перебора. Такой подход возможен благодаря сравнительно небольшому количеству признаков в данной задаче и диктуется требованиями экспертов.

Так как количество признаков в искомом наборе $\mathcal{A}$ неизвестно, множество индексов объектов $\mathcal{I}$ разбивается случайным образом на два подмножества, $\mathcal{I} = \mathcal{L}\sqcup\mathcal{T}$, обучающее и тестовое. Параметры $\mathbf{w}$ оцениваются на подвыборке $D_\mathcal{L}$, а качество прогноза вычисляется

на подвыборке $D_{\mathcal{T}}$. Максимальное число признаков при решении задачи фиксировано экспертами: $|\mathcal{A}|$ не должна превышать четырех. Наборы признаков, полученные в результате решения задачи (8), будем называть оптимальным, а сами признаки — наиболее информативными.

Оценка объема выборки

Данные, использованные при проведении вычислительного эксперимента, содержат признаковые описания пациентов, принадлежащих одному из классов: больные, перенесшие инфаркт или имеющие предрасположенность к инфаркту. В качестве признаков (биомаркеров) используются концентрации белков и их соединений, абсорбированные на поверхности кровяных телец. В классах содержится четырнадцать и сорок объектов соответственно. При таком объеме данных возникает задача оценки минимального объема выборки m^*, необходимого для получения статистически достоверных результатов классификации. В данном разделе рассмотрены четыре способа оценки объема выборки. Результаты оценки объема выборки описаны и проанализированы в разделе «Вычислительный эксперимент».

Метод доверительных интервалов. Рассмотрим выборку, в который каждый объект описывается одним признаком $D = \{(x_i, y_i)\}, i \in \mathcal{I} = \{1, \ldots, m\}$. Пусть $\Delta = \bar{x} - \mu$ — разница между средним арифметическим

$$\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i$$

и известным математическим ожиданием μ. При известном среднеквадратическом отклонении σ случайная величина принадлежит страндартному нормальному распределению

$$Z = \frac{\bar{x} - \mu}{\sigma}\sqrt{m} = \frac{\Delta}{\sigma}\sqrt{m} \sim \mathcal{N}(0, 1). \tag{9}$$

Тогда

$$\Delta = z_{\alpha/2}\frac{\sigma}{\sqrt{m}},$$

где $z_{\alpha/2}$ таково, что $P\left\{|Z| \geqslant z_{\alpha/2}\right\} = \alpha$. Отсюда получаем формулу для оценки размера выборки

$$m^* = \left(\frac{z_{\alpha/2}\sigma}{\Delta}\right)^2. \tag{10}$$

При $m \geqslant 30$ можно пользоваться предположением о нормальности случайной величины Z, если величины x_i распределены ненормально, а также при неизвестном σ^2, заменив его в выражении (9) на

$$s = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2}.$$

Однако в случае $m \leqslant 30$ для использования этой формулы необходимо, чтобы случайные величины x_i были распределены нормально; кроме того, среднеквадратическое отклонение σ должно быть известно.

В данной работе рассматривается многопризнаковая задача, однако в предположении, что все признаки из наиболее информативных наборов взаимно независимы, формула (10) верна для каждого из них. Вычисляя объем выборки для различных признаков, будем

получать различные значения. Для получения общей оценки можно взять среднее или наибольшее из них. При таком подходе можно получить лишь грубую оценку, так как более правдоподобна альтернативная гипотеза о том, что распределение признаков в выборке представляет собой смесь из двух нормальных распределений:

$$x_i \sim \begin{cases} \mathcal{N}(\mu_1, \sigma_1^2), & \text{с вероятностью } p_i; \\ \mathcal{N}(\mu_2, \sigma_2^2), & \text{с вероятностью } 1 - p_i, \end{cases} \tag{11}$$

где параметр p_i задан с помощью (4)

Метод оценки объема выборки в логистической регрессии. Фиксируем некоторое множество $\mathcal{A}$ индексов признаков, используемых для получения прогноза. Для каждого из признаков, вошедших в этот набор, можно оценить объем выборки, необходимый чтобы включить этот признак в набор. Для этого рассмотрим нулевую гипотезу вида

$$H_0 : w_j = 0,\ j \notin \mathcal{A},$$

где w_j — j-тая компонента вектора параметров $\mathbf{w}$ логистической регрессии. Таким образом, нулевая гипотеза заключается в предположении, что j-тый признак не включается в модель. Оценив вектор параметров при нулевой гипотезе, получим $\mathbf{w}_{\mathcal{A}}$, а при принятии альтернативной гипотезы — $\mathbf{w}_{\mathcal{A}^*}$, где множество $\mathcal{A}^*$ получается из $\mathcal{A}$ добавлением к нему индекса j. Тогда нулевая и альтернативная гипотезы могут быть переформулированы относительно параметров θ_i бернуллиевского распределения $\mathcal{B}(\theta)$ и представлены в виде

$$H_0 : \boldsymbol{\theta} = \boldsymbol{\theta}_{\mathcal{A}}, H_1 : \boldsymbol{\theta} = \boldsymbol{\theta}_{\mathcal{A}^*}.$$

При этом неважно, какие именно значения принимают θ_i в каждом случае, интерес представляет только пороговое значение функции регрессии θ_0. Окончательно сформулируем гипотезы в виде:

$$H_0 : 1 - {}_0 = p_0, H_1 : 1 - {}_0 = p_1.$$

Выберем в качестве тестовой статистики

$$Z = \frac{\hat{p} - p_0}{\sqrt{p_0 q_0 / m}}, \quad \hat{p} = \frac{1}{m} \sum_{i=1}^{m} y_i$$

где $\hat{p}$ — оценка максимального правдоподобия для параметра θ. При принятии альтернативной гипотезы статистика Z имеет нормальное распределение

$$Z \sim \mathcal{N}\left(p_1 - p_0, \sqrt{\frac{p_1 q_1}{p_0 q_0}}\right).$$

Тогда величина

$$Z\sqrt{\frac{p_0 q_0}{p1 q_1}} + \frac{p_0 - p_1}{\sqrt{p_1 q_1 / m}} = \sqrt{\frac{p_0 q_0}{p_1 q_1}} \left(Z + \frac{p_0 - p_1}{\sqrt{p_0 q_0}} \sqrt{m}\right) \sim \mathcal{N}(0, 1).$$

Выбрав уровень значимости α и мощность критерия $1 - \beta$, запишем выражение для мощности

$$1 - \beta = P\{|Z| > Z_{\alpha/2} | H_1\} = \Phi\left(\sqrt{\frac{p_0 q_0}{p_1 q_1}} \left(Z_{\alpha/2} + \frac{p_0 - p_1}{\sqrt{p_0 q_0 / m}}\right)\right).$$

Тогда необходимый объем выборки вычисляется по формуле

$$m^* = \frac{p_0 q_0 \left(Z_{1-\alpha/2} + Z_{1-\beta} \sqrt{\frac{p_1 q_1}{p_0 q_0}} \right)^2}{(p_1 - p_0)^2}. \tag{12}$$

Заметим, что вычисленный объем выборки зависит от номера признака, относительно которого сформулирована нулевая гипотеза.

Скользящий контроль. Метод скользящего контроля позволяет оценить необходимый объем выборки с точки зрения контроля над переобучением. При таком подходе производится разбиение выборки на две непересекающиеся подвыборки: обучающую и тестовую. Пусть $\mathcal{I} = \{1, \ldots, m\}$ — множество индексов объектов выборки, разобьем его на два непересекающихся подмножества $\mathcal{I} = \mathcal{L} \bigsqcup \mathcal{T}$. Тогда обучающей выборкой назовем $D_{\mathcal{L}} = \{(\mathbf{x}_i, y_i)\}, i \in \mathcal{L}$, а тестовой — $D_{\mathcal{T}} = \{(\mathbf{x}_i, y_i)\}, i \in \mathcal{T}$. Фиксируем набор признаков с индексами $\mathcal{A}$. Уменьшение функционала качества $S(\mathcal{A}, D_{\mathcal{T}})$ на тестовой выборке по сравнению с его значением $S(\mathcal{A}, D_{\mathcal{L}})$ на обучающей выборке свидетельствует о наличии переобучения. Переобучением назовем отношение

$$\mathrm{RS}(m) = \frac{S(\mathcal{A}, D_{\mathcal{T}(m)})}{S(\mathcal{A}, D_{\mathcal{L}(m)})}. \tag{13}$$

В этом случае модель f хорошо описывает данные, на которых она была настроена, но плохо приближает тестовую выборку. Переобучение может являться следствием недостаточного объема выборки. Чтобы оценить необходимый объем выборки, будем последовательно наращивать объем выборки m, производя разбиение на обучение и контроль в заданном отношении: $|\mathcal{T}(m)|/|\mathcal{L}(m)| = \mathrm{const} \leqslant 0,5$. Для каждого значения m будем вычислять отношение (13). При увеличении объема выборки оно стремится к единице. Будем считать, что объем выборки m^* достаточен, если начиная с него величина RS не меньше чем заданное $1 - \varepsilon_1$.

Оценка объема выборки с использованием расстояния Кулльбака-Лейблера. Предлагаемый подход основан на вычислении расстояния между функциями распределения параметров регрессионной модели. Рассмотрим некоторое множество индексов объектов $\mathcal{B}_1 \in \mathcal{J}$, а также множество $\mathcal{B}_2 \in \mathcal{J}$, такое что:

$$|\mathcal{B}_1 \setminus \mathcal{B}_2 \cup \mathcal{B}_2 \setminus \mathcal{B}_1| \leqslant 2.$$

Таким образом, множество $\mathcal{B}_2$ может быть получено из $\mathcal{B}_1$ путем удаления, добавления или замены одного элемента. Оценивая параметры на различных подвыборках, будем получать различные результаты. На рисунке 1 продемонстрировано, как меняется положение разделяющей гиперплоскости, определяемой выражением

$$\mathbf{x}^T \mathbf{w} = \ln(\frac{c_0}{1 - c_0})$$

при добавлении в выборку двух элементов. Если объем выборки $D_{\mathcal{B}_1}$ достаточно велик, небольшое изменение ее состава $D_{\mathcal{B}_2}$ не должно приводить к существенному изменению параметров. Простейший способ сравнивать параметры на различных подвыборках — с помощью

$$||\mathbf{w}_1 - \mathbf{w}_2|| = \sqrt{\sum_{i=1}^{|\mathcal{A}|} \left(w_i^1 - w_i^2\right)^2}.$$

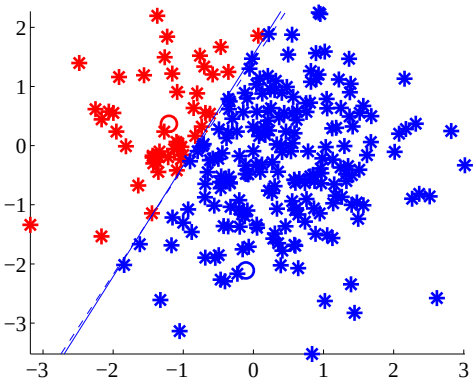


Рис. 1. Два класса, разделенные гиперплоскостью. Пунктирной линией обозначено положение гиперплоскости после того как два новых объекта (выделенных окружностями), были добавлены в выборку.

Предлагается сравнивать функции распределения параметров модели на подвыборках $D_{\mathcal{B}_1}$ и $D_{\mathcal{B}_2}$ с помощью расстояния Кулльбака-Лейблера.

Рассмотрим модель (4) и предположение о распределении случайной величины y_i (1). Зафиксировав выборку D и модель $f_{\mathcal{A}} = f(X_{\mathcal{A}}^T \mathbf{X})$, перепишем (1) в виде

$$p(\mathbf{y}|X, \mathbf{w}, f_{\mathcal{A}}) \equiv p(D|\mathbf{w}, f_{\mathcal{A}}) = \prod_{i=1}^{m} \theta_i^{y_i} (1-\theta_i)^{1-y_i}. \tag{14}$$

Предположим также, что вектор параметров $\mathbf{w}$ регрессии имеет нормальное распределение $\mathbf{w} \sim \mathcal{N}(\mathbf{w}_0, \sigma^2 I_{|\mathcal{A}|})$ с плотностью

$$p(\mathbf{w}|f_{\mathcal{A}}, \alpha) = \left(\frac{\alpha}{2\pi}\right)^{\frac{|\mathcal{A}|}{2}} \exp(-\frac{\alpha}{2}||\mathbf{w} - \mathbf{w}_0)||^2), \tag{15}$$

где $\alpha^{-1} = \sigma^2$, а $I_{|\mathcal{A}|}$ — единичная матрица размерности $|\mathcal{A}|$.

Найдем плотность распределения $p(\mathbf{w}|D, \alpha, f_{\mathcal{A}})$ параметров модели, воспользовавшись формулой Байеса

$$p(\mathbf{w}|D, \alpha, f_{\mathcal{A}}) = \frac{p(D|\mathbf{w}, f_{\mathcal{A}}) p(\mathbf{w}|\alpha, f_{\mathcal{A}})}{p(D|\alpha, f_{\mathcal{A}})}, \tag{16}$$

где $p(D|\mathbf{w}, f_{\mathcal{A}})$ — правдоподобие данных, $p(\mathbf{w}|\alpha, f_{\mathcal{A}})$ — задаваемая априорно плотность распределения параметров модели. В выражении (16) нормировочный множитель $p(D|\alpha, f_{\mathcal{A}})$ определяется выражением

$$p(D|\alpha, f_{\mathcal{A}}) = \int p(D|\mathbf{w}, f_{\mathcal{A}}) p(\mathbf{w}|\alpha, f_{\mathcal{A}}) d\mathbf{w}.$$

Подставив (14) и (15) в (16) и обозначив $Z(\alpha) = p(D|\alpha, f_{\mathcal{A}})$, получим

$$p(\mathbf{w}|D, f_{\mathcal{A}}) = \frac{p(y|\mathbf{x}, \mathbf{w}, f_{\mathcal{A}}) p(\mathbf{w}|f_{\mathcal{A}}, \alpha)}{Z(\alpha)} =$$

$$= \frac{\alpha^{\frac{|\mathcal{A}|}{2}}}{(2\pi)^{\frac{|\mathcal{A}|}{2}} Z(\alpha)} \exp(-\frac{\alpha}{2}||\mathbf{w} - \mathbf{w}_0)||^2) \prod_{i=1}^{m} \theta_i^{y_i} (1-\theta_i)^{1-y_i},$$

где $Z(\alpha)$ — нормировочный множитель.

Пусть имеются две выборки, $D_{\mathcal{B}_1}$ и $D_{\mathcal{B}_2}$. Обозначим соответствующие апостериорные распределения $p_1(\mathbf{w}) \equiv p(\mathbf{w}|D_{\mathcal{B}_1}, \alpha, f_{\mathcal{A}})$ и $p_2(\mathbf{w}) \equiv p(\mathbf{w}|D_{\mathcal{B}_2}, \alpha, f_{\mathcal{A}})$. Для оценки сходства плотностей распределения параметров вычислим расстояние Кулльбака-Лейблера между ними

$$D_{\mathrm{KL}}(p_1, p_2) = \int_{\mathbf{w}\in\mathcal{W}} p_1(\mathbf{w}) \frac{p_1(\mathbf{w})}{p_2(\mathbf{w})} d\mathbf{w}. \tag{17}$$

Используя в качестве меры изменения распределения $p(\mathbf{w}|D, f_{\mathcal{A}})$ расстояние Кулльбака-Лейблера, оценим необходимый объем выборки. Для этого будем N раз случайным образом удалять из выборки по одному объекту, уменьшая ее размер, и вычисляя каждый раз плотность распределения вектора $\mathbf{w}$ с помощью (15). Затем посчитаем расстояние (17) между «соседними» распределениями, т.е. между функциями распределения параметров, которые оценивались на подвыборках, отличающихся друг от друга только одним объектом. Проведем эту процедуру N раз и усредним полученные расстояния. Считаем объем выборки m^* достаточным, если начиная с расстояние (17) меняется не больше чем на заранее заданное число ε_2.

Результаты вычислительного эксперимента

Эксперимент на реальных данных. В данном разделе описан вычислительный эксперимент, который проводился на данных лаборатории анализа крови «Иммуноклин». Данные содержат измерения концентрации 20-ти белков и их соединений на поверхности кровяных телец пациентов двух классов, содержащих 31 и 14 объектов соответственно. В таблице 2 приведен список исследуемых биомаркеров с их порядковыми номерами.

Таблица 1. Результаты выбора признаков

$\mathcal{A}$	$S(\mathcal{A})$
K, L , L/P	0.9750
K, L, K/M, K/Q	0.9671
K, L, L/M, L/T/SO	0.9933
K, L, K/M, L/R	0.9867
K, K/M, L/P,	0.9742

В таблице 1 указаны наборы маркеров, доставивших наибольшие значения максимизируемому критерию AUC и сами значения этого критерия. Для исследования были выбраны K лучших наборов.

В данном случае выбрано значение $K = 5$.

Таблица 2. Число вхождений признаков в K оптимальных наборов для каждого признака.

K	L	K/M	L/M	K/N	K/O	L/O	K/P	L/P	K/Q
5	4	3	1	0	0	0	0	2	1

K/R	L/R	L/R/SA	L/T/SA	L/T/SO	U/V	U/W	U/X	U/Y	U/Z
0	1	0	0	1	0	0	0	0	0

Одной из важный практических задач, решаемых в рамках проводимых исследований, является задача снижения стоимости клинического исследования одного пациента, решаемая путем уменьшения числа измеряемых биомаркеров. Предложено измерять только наиболее информативные биомаркеры, выбранные следующим образом. Объединив признаки

из всех наборов из колонки «$\mathcal{A}$» таблицы 1, получим множество наиболее информативных признаков $\mathcal{S} = \bigcup_{i=1}^{K} \{\mathcal{A}_i\}$. Для каждого признака подсчитано количество его вхождений в это множество. Таблица 2 показывает число вхождений каждого биомаркера в $\mathcal{S}$.

Оценка необходимого объема выборки

На гистограмме 2 отложены оценки объема выборки m^*, вычисленные по формулам (10) и (12), от признака. Заметим, что нет необходимости проводить усреднение, как это предлагалось в разделе «Метод доверительных интервалов», по всем признакам, так как в модель вошли лишь некоторые из них, остальные являются неинформативными и учитываться не должны.

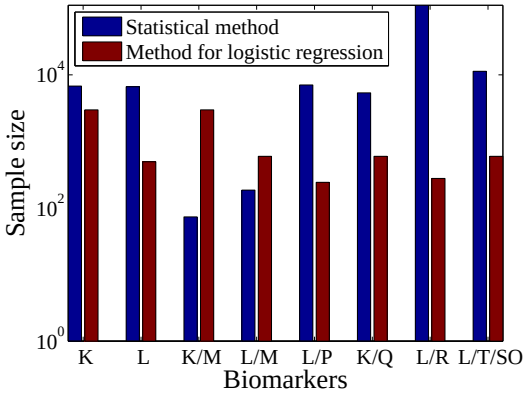


Рис. 2. Оценка объема выборки, полученная с помощью метода доверительных интервалов и с помощью метода для логистической регрессии, для каждого из признаков.

Заметим, что зависимости, изображенные на рисунке 2 носят одинаковый характер: положения максимумов и минимумов практически совпадают. Это происходит от того, что размер выборки, оцененный для j-го признака, зависит от информативности этого признака. В логистической регрессии такие признаки имеют большое по абсолютной величине значение соответствующего элемента вектора параметров w_j. В формуле (12) в знаменателе стоит квадрат разности $(c_0 - \sigma_1)^2$. Чем ближе к нулю величина w_j, тем меньше $(c_0 - \sigma_1)^2$, и больше m^*. Таким образом, наименьшие значения объема выборки соответствуют наиболее информативным признакам, а аномально большие (порядка 10^4 и выше) наблюдаются у признаков, которые в модель не входят — у них w_j близко к нулю.

На рисунке 3 изображена зависимость величины $\mathrm{RS}(m)$, определяемой (13) от размера выборки, на которой проводился скользящий контроль. При данном размере выборки функция $\mathrm{RS}(m)$ не успевает выйти на асимптоту, и о дальнейшем поведении функции по рисунку 3 судить нельзя, поэтому метод скользящего контроля дает оценку на необходимый объем выборки $m^* \geqslant 30$.

На рисунке 4 изображены зависимости евклидового расстояния между параметрами и расстояния Кулльбака-Лейблера между их плотностями, усредненного по $N = 100$ разбиениям, от количества объектов в выборке. Видно, что при $m \geqslant 25$ оба графика меняется достаточно плавно. Таким образом, минимальный объем выборки, оцененный с помощью расстояния Кулльбака-Лейблера $m^* \leqslant 30$.

Для сравнения перечисленных методов, приведем таблицу 3, содержащую оценки необходимого объема выборки для каждого из рассмотренных методов. Особенностью исполь-

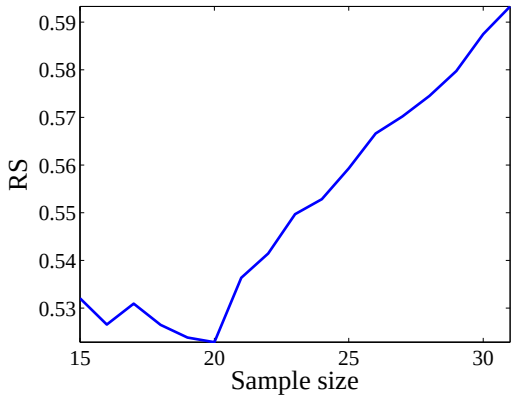
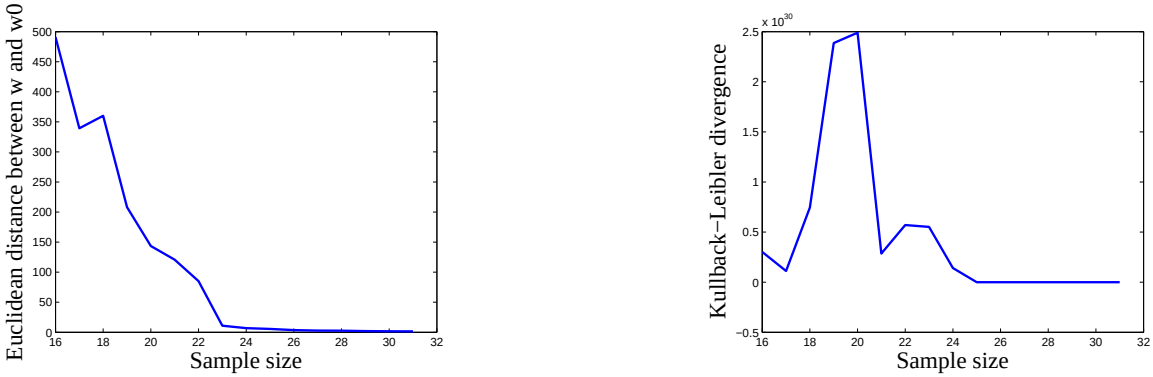


Рис. 3. Оценка объема выборки, полученная с помощью метода Демиденко, для каждого из признаков.



a. b.

Рис. 4. a. Усредненное еклидово расстояние между параметрами модели, $||\mathbf{w}_m - \mathbf{w}_{m+1}||$ b. Усредненное расстояние Кулльбака-Лейблера.

Таблица 3. Результаты оценки необходимого объема выборки с помощью четырех различных методов.

Метод доверительных интервалов	Демиденко	Скользящий контроль	Расстояние Кулльбака-Лейблера
$10^2 - 10^4$	$\sim$50	$\geqslant 30$	$\geqslant 30$

зованных при проведениии вычислительного эксперимента данных является небольшой объем выборки, поэтому метод скользящего контроля и расстояние Кулльбака-Лейблера дают лишь нижнюю оценку, так как эти методы больше подходят для оценки достаточного объема выборки, то есть применимы когда объем выборки слишком велик. Метод доверительных интервалов и метод Демиденко дают результаты, численно отличающиеся на порядки, однако качественно схожие друг с другом. Последнее объясняется тем, что объем выборки, оцениваемый этими методами, зависит от информативности признака. Различие можно объяснить грубостью предположений, сделанных в рамках метода доверительных интервалов.

Эксперимент на синтетических данных. Работа алгоритма также была протестирована на примере с искусственными данными, их структура отображена на рисунке 5.

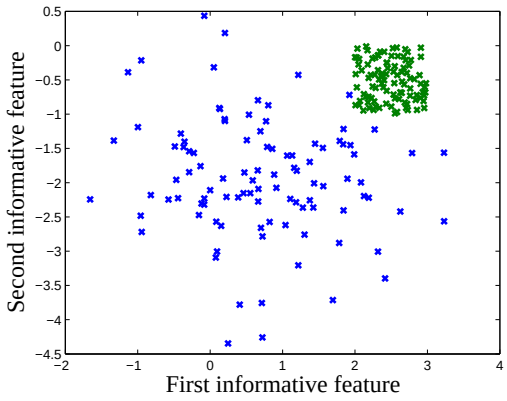


Рис. 5. Распределение данных в пространстве двух информативных признаков.

Оба класса имеют по один шумовой и два информативных признака и содержат по 100 объектов.

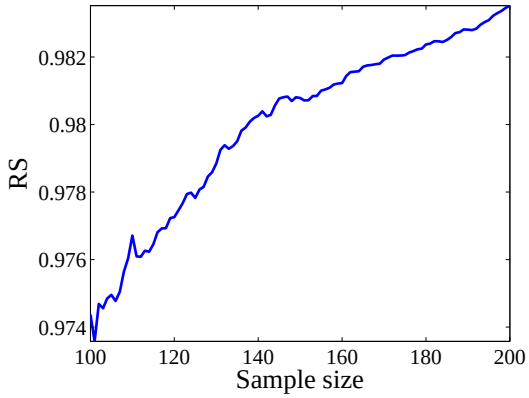


Рис. 6. Зависимость RS(m) при разделении выборки на обучающую и контрольную в отношении 3:1.

Из рисунка 6 видно, что уже начиная с $m = 100$ величина RS(m) меняеется не более чем на 0.001, т.е. можно считать, что $m^* \leqslant 100$.

На гистограмме 7 отражены результаты оценки m^* методом доверительных интервалов и методом для логистической регрессии. При этом менее точный метод доверительных интервалов дает результат, более близкий к m^*, полученному с помощью методов скользящего контроля и расстояния Кулльбака-Лейблера. Дело в том, что распределение рассматриваемых данных сильно отличается от предполагаемого расределения (11) реальных данных, описанных в разделе 12. Рассмотрим выборку, с признаковым описанием, состоящим из одной случайной величины, распределенной как (11). Меняя расстояние $|\mu_1 - \mu_2|$ между математическими ожиданиями компонент смеси, будем наблюдать за результатами оценки m^*, полученными с помощью методов скользящего контроля и логистической регрессии. Результаты приведены на гистограмме 8

В этом случае метод скользящего контроля дает завышенные результаты, в то время как метод для логистической регрессии оказывается более точен.

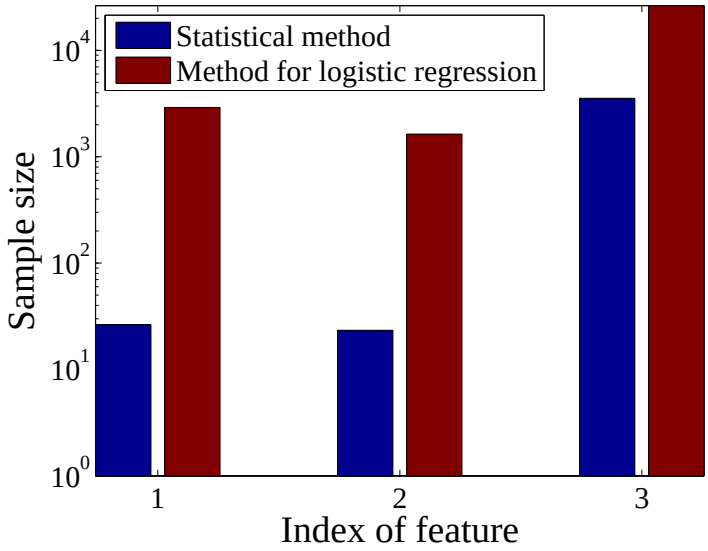


Рис. 7. Оценка объема выборки, полученная с помощью метода доверительных интервалов и с помощью метода для логистической регрессии, для каждого из признаков.

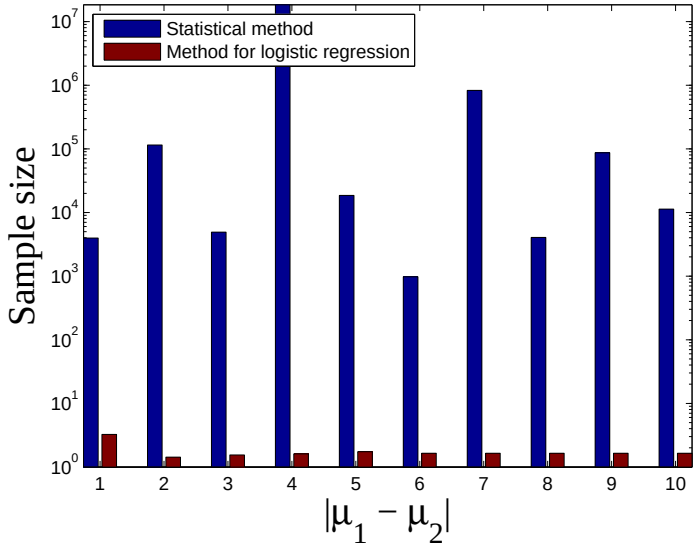


Рис. 8. Оценка объема выборки, полученная с помощью метода доверительных интервалов и с помощью метода для логистической регрессии, для для различных значений $|\mu_1 - \mu_2|$.

Заключение

В работе описан алгоритм прогнозирования вероятности наступления инфаркта пациентов; описан способ оценки параметров и выбора наиболее информативных признаков методом полного перебора. Этот подход возможен благодаря небольшому количеству признаков и дает экспертам гарантию, что выбран оптимальный набор. Описаны способы получения оценки необходимого объема выборки пациентов. Предложен новый метод оценки объема выборки, основанный на вычислении расстояния Кулльбака-Лейблера между плотностями распределения параметров модели, оцениваемыми при различных разбиениях выборки. Все методы протестированы на реальных и синтетических данных.

Литература

[1] *Hosmer D., Lemeshow S.* Applied logistic regression. N. Y.: Wiley, 2000. 375 p.

[2] *Bishop C. M.* Pattern recognition and machine learning. Springer, 2006. 738 p.

[3] *MacKay D. J. C.* Information theory, inference, and learning algorithms. Cambridge University Press, 2003. 628 p.

[4] *Friedman J., Hastie, Tibshirani R.* Additve logistic regression: a statistical way of boosting // The Annals of Statistics. 2000. V. 28, № 2. P. 337–407.

[5] *Madigan D., Rideway G.* Discussion of least square regression. В сб. Efron B. [et al.]. Least Angle Regression // The Annals of Statistics. 2004. V. 32, № 2. P. 465–469.

[6] *Fawcet T.* ROC graphs: notes and practical considerations for researchers // HP Laboratories, 2004. 38 p.

[7] *Реброва О. Ю.* Статистический анализ медицинских данных. Применение прикладного пакета Statistica. М.: Медиасфера, 2002. 312 с.

[8] *Demidenko E.* Sample size determination for logistic regression revisited // Statist. Med. 2007; 26:3385–3397.

[9] *Rosner B.* Fundamentals of biostatistics. Duxbury Press, 1999. 816 p.

[10] *Bos S.* How to partition examples between cross-validation set and fraining set? / Saitama, Japan: Laboratory for information representation RIKEN. 1995. 4 p.

[11] *Amari S., Murata N., Muller K.-R., Finke M., Yang H.H.* Asymptotic statistical theory of overtraining and cross-validation. // IEEE Transactions on Neural Networks, 1997. V. 8, No. 5. P. 985–996.

[12] *Perez-Cruz F.* Kullback-Leibler divergence estimation of continuous distributions // IEEE International Symposium on Information Theory, 2008.

[13] Standart flow cytometry analysis of nondental patients. Paris: ImmunoClin laboratory. 2007. 1 p.

Локальные методы прогнозирования с выбором метрики*

А. А. Варфоломеева
annette92@mail.ru
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В данной работе рассматривается локальный метод прогнозирования временных рядов. Исследуется вопрос выбора функции близости для нахождения похожих участков ряда. Проводится сравнение эффективности алгоритма построения прогноза при использовании различных метрик на модельных данных и временных рядах потребления электроэнергии и цен на сахар.

Ключевые слова: *локальное прогнозирование, функция близости, функционал качества, k ближайших соседей.*

Введение

Методы построения прогноза временных рядов делятся на глобальные (использующие всю предысторию ряда) и локальные (используют только её часть). В общем случае прогнозирования рядов решается задача поиска оптимальных парамеров алгоритма: число ближайших соседей, длина предыстории и выбор функции близости. В данной работе рассматривается локальный метод прогнозирования, основанный на алгоритме поиска k ближайших соседей, который был описан в работах Дж. Макнеймса [1] и Ю.И. Журавлева [3]. В работе В.П. Федоровой [5] подробно рассмотрен вопрос оптимизации некоторых параметров метода прогнозирования, основанного на алгоритме "k ближайших соседей". В настоящей работе изучается проблема поиска метрики, дающей наибольшую эффективность данного алгоритма. Рассматривая набор определенных метрик и минимизируя функционал ошибки алгоритма, выбирается наиболее подходящая под данный временной ряд метрика.

Постановка задачи

Рассматриваются одномерные временные ряды, т.е. такие, в которых значением ряда в каждый момент времени является вещественное число. Задача прогнозирования временного ряда состоит в том, чтобы по известному отрезку временного ряда

$$(f_1, f_2, \ldots, f_n)$$

предсказать следующие t его значений:

$$(f_{n+1}, f_{n+2}, \ldots, f_{n+t}).$$

Решается задача построения локального метода прогнозирования временных рядов, основанного на алгоритме "ближайших соседей". Для оценки степени близости объектов предлагается использовать различные метрики и сравнить качество прогноза при их использовании. В данной работе длина предыстории считается фиксированной и равной l. Алгоритм "ближайших соседей" состоит из следующих этапов:

Научный руководитель В. В. Стрижов

1. Найти в предыстории среди всех векторов размерности l, составленных из отрезков временного ряда $(f_i, f_{i+1}, \ldots, f_{i+l-1})$, k векторов, наиболее похожих после линейных преобразований на вектор $(f_{n-l+1}, f_{n-l+2}, \ldots, f_n)$. При этом мера сходства определяется с помощью одной из рассматриваемых в работе метрик.
2. Пусть $\{(f_{i_1-l+1}, \ldots, f_{i_1}), \ldots, (f_{i_k-l+1}, \ldots, f_{i_k})\}$ — k ближайших соседей для предыстории $(f_{n-l+1}, \ldots, f_n)$. Прогноз $(\hat{f}_{n+1}, \hat{f}_{n+2}, \ldots, \hat{f}_{n+t})$ вычисляется как взвешенное среднее арифметическое k векторов:

$$\{(f_{i_1+1}, \ldots, f_{i_1+t}), \ldots, (f_{i_k+1}, \ldots, f_{i_k+t})\}.$$

Данная работа посвящена анализу различных метрик и зависимости качества прогноза от выбора одной из них. Рассматривается набор метрик P:

— Стандартная Евклидова метрика:

$$\rho_E(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})}. \tag{1}$$

— Диагонально взвешенная Евклидова метрика:

$$\rho_{wE}(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^T\Lambda^2(\mathbf{x}-\mathbf{y})}, \text{ где } \Lambda = \mathrm{diag}(\lambda). \tag{2}$$

— Метрика Минковского L_p:

$$\rho_{L_p}(\mathbf{x}, \mathbf{y}) = (\sum_i |x_i - y_i|^p)^{1/p}, \text{ где } p \in \mathbb{N}. \tag{3}$$

Для оценки качества алгоритма используется функционал ошибки SMAPE(Symmetric Mean Absolute Percent):

$$\mathrm{SMAPE}(f, \hat{f}, n, t) = \frac{1}{t}\sum_{i=1}^{t} \frac{|\hat{f}_{n+i} - f_{n+i}|}{|\hat{f}_{n+i} + f_{n+i}|/2} * 100\%. \tag{4}$$

Этот функционал учитывает отклонение прогноза от точного значения относительно величины значения. Тогда задача прогнозирования состоит в поиске такого вектора $(\hat{f}_{n+1}, \hat{f}_{n+2}, \ldots, \hat{f}_{n+t})$, что

$$(\hat{f}_{n+1}, \hat{f}_{n+2}, \ldots, \hat{f}_{n+t}) = \arg\min_{(\hat{f})^\rho} SMAPE(f, (\hat{f})^\rho, n, t), \tag{5}$$

где $(\hat{f})^\rho = ((\hat{f}_{n+1})^\rho, (\hat{f}_{n+2})^\rho, \ldots, (\hat{f}_{n+t})^\rho)$ — прогноз, вычисленный с использованием одной из рассматриваемых метрик $\rho \in P$.

Описание алгоритма

Опишем более подробно используемый алгоритм. Для начала напомним основные обозначения:

— $\mathbf{S} = (f_1, f_2, \ldots, f_N)$ — известный временной ряд.
— l — длина предыстории, считается фиксированной.
— t — длина прогнозируемого отрезка.
— $\rho(\mathbf{x}, \mathbf{y}) \in P$ — функция близости между векторами $\mathbf{x}$ и $\mathbf{y}$.

Основная процедура. Пусть, для определенности, прогнозируются последние t значений временного ряда $\mathbf{S}^{T\times 1}$:

$$(f_{N-t+1}, f_{N-t+2}, \ldots, f_N).$$

Тогда за известный отрезок временного ряда принимается $(f_1, f_2, \ldots, f_n)$, где $n = N - t$. Для начала выделим предысторию прогнозируемого вектора $\mathbf{y} = (f_{n-l+1}, f_{n-l+2}, \ldots, f_n)$ и составим матрицу $\mathbf{X}^{l\times(n-t)}$, состоящую из всех векторов $\mathbf{x}_i = (f_i, f_{i+1}, \ldots, f_{i+l-1})$ размерности l, входящих в известный временной ряд $\mathbf{S}$, и имеющих после себя как минимум t известных значений: они рассматриваются как потенциальные соседи. Далее введем матрицу $\mathbf{D}^{4\times(n-t)}$, в которую для выбранной функции близости и каждого потенциального соседа $\mathbf{x}_i$ запишем его характеристики:

— индекс первого элемента вектора $\mathbf{x}_i$ i;
— расстояние до предыстории $\rho(\mathbf{x}_i, \mathbf{y})$, вычисленное с помощью определенной метрики;
— параметры линейного преобразования вектора при вычислении расстояния $\rho(\mathbf{x}_i, \mathbf{y})$.

Отсортируем матрицу $\mathbf{D}$ по величине расстояния $\rho(\mathbf{x}_i, \mathbf{y})$ и выделим первые k векторов $\mathbf{x}_i$, сооответствующие ближайшим соседям. Найдем для каждого ближайшего соседа $\mathbf{x}_i$ его продолжение $\mathbf{k}_i = (f_{i+l}, f_{i+l+1}, \ldots, f_{i+l+t-1})$ и запишем их в матрицу $\mathbf{K}^{t\times k}$. Также вычисляем веса W_i, с которыми каждый из векторов $\mathbf{k}_i$ учитывается при построении прогноза: они зависят от расстояния до предыстории $\mathbf{y}$ по формуле, предложенной в [4]:

$$W_i = \left(1 - \left(\frac{\rho(\mathbf{k}_i, \mathbf{y})}{\rho(\mathbf{k}_{k+1}, \mathbf{y})}\right)^2\right)^2, \tag{6}$$

где $\rho(\mathbf{k}_i, \mathbf{y})$ — расстояние до i-го ближайшего соседа. Для нормировки весов W_i, вычислим их общую сумму и поделим каждый из весов на нее:

$$w_i = \frac{W_i}{\sum_{j=1}^{t} W_j}.$$

Далее строится прогноз $\hat{\mathbf{x}}_{n+1} = (\hat{f}_{n+1}, \hat{f}_{n+2}, \ldots, \hat{f}_{n+t})$ как взвешенное среднее арифметическое найденных отрезков:

$$\hat{\mathbf{x}}_{n+1} = \sum_{i=1}^{t} w_i * \mathbf{k}_i, \tag{7}$$

и записывается в продолжение данного временного ряда:

$$\hat{S} = (f_1, f_2, \ldots, f_n, \hat{f}_{n+1}, \hat{f}_{n+2}, \ldots, \hat{f}_{n+t}). \tag{8}$$

Вычисление расстояния между отрезками. Во всех исследуемых метриках похожие отрезки ищутся с точностью до линейного преобразования: $\tilde{\mathbf{x}} = a * \mathbf{x} + b$, где $a, b \in \mathbb{R}$. Следовательно расстояние между векторами $\mathbf{x}$ и $\mathbf{y}$ определяется как:

$$\rho(\mathbf{x}, \mathbf{y}) = \min_{a,b} \rho(\tilde{\mathbf{x}}, \mathbf{y}), \tag{9}$$

при этом $\rho \in P$.

Для диагонально взвешенной Евклидовой метрики общим предположением является то, что конец предыстории для прогноза более важен, чем его начало, поэтому параметры

λ_{ii} увеличиваются с порядковым номером i. В работе предполагается, что последовательность весовых параметров λ_{ii} имеет степенной вид:

$$\lambda_{ii} = \lambda^{t-i+1}, 0 < \lambda_i \leqslant 1, i = 1, \ldots, t. \tag{10}$$

Следовательно, требуется задать, например, $\lambda_{11} = \lambda$.

Вычислительный эксперимент

Алгоритм протестирован на модельных и реальных данных. В качестве модельных данных взят ряд, образованный с помощью функции

$$f_1(t) = sin(t) * cos(0.01t) \text{ , где } t = 1, 2, \ldots, 1000.$$

Считаем известными первые 800 точек ряда и строим по ним прогноз длиной $t = 200$. Для исследуемых модельных данных $f_1(t)$ зафиксируем длину каждого соседа $l = 80$. В качестве реальных временных рядов использованы данные о почасовом потреблении электроэнергии $f_2(t)$ (рис.1) и данные о ценах на сахар $f_3(t)$ (рис.2). Первый ряд является строго периодичным и почти не содержит шумов, тогда как второй является зашумленным. В случае реальных данных $f_2(t)$ и $f_3(t)$ установим длину соседа $l = 60$ и будем прогнозировать последние $t = 400$ значений. На всех графиках, отражающих построение прогноза, зеленым цветом выделены линии, полученные с помощью используемого в работе алгоритма, синим — линии, построенные по известному временному ряду, а красные линии отражают поточечную разницу между прогнозом и известным значением ряда.

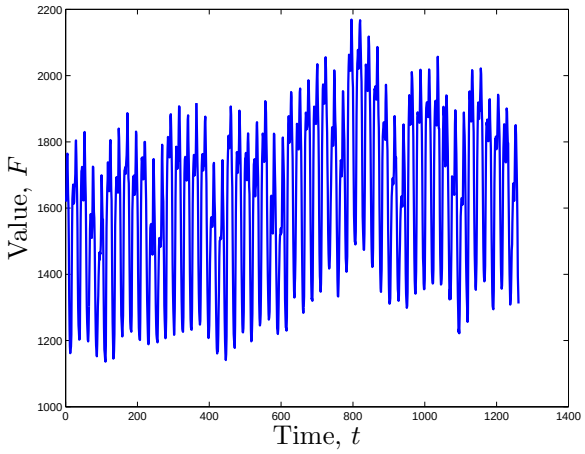


Рис. 1. Вид данных о потреблении электроэнергии.

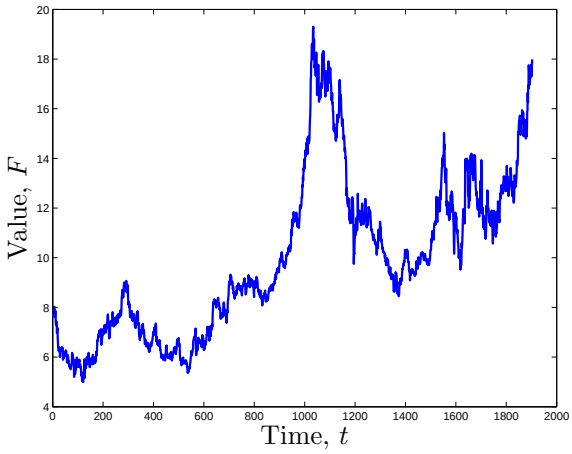


Рис. 2. Вид данных о ценах на сахар.

Стандартная Евклидова метрика.

Рассмотрим качество работы алгоритма, использующего стандартную Евклидову метрику (1). Исследуемым параметром для оптимизации в данном случае служит количество ближайших соседей k. Зависимость величины ошибки SMAPE (4) от количества ближайших соседей k и построение прогноза для рассматриваемых рядов отражены на рис.3, рис.4 и рис.5.

Диагонально взвешенная Евклидова метрика. При использовании метрики (2) необходимо оптимизировать также и весовые параметры метрики λ_{ii}, которые задаются

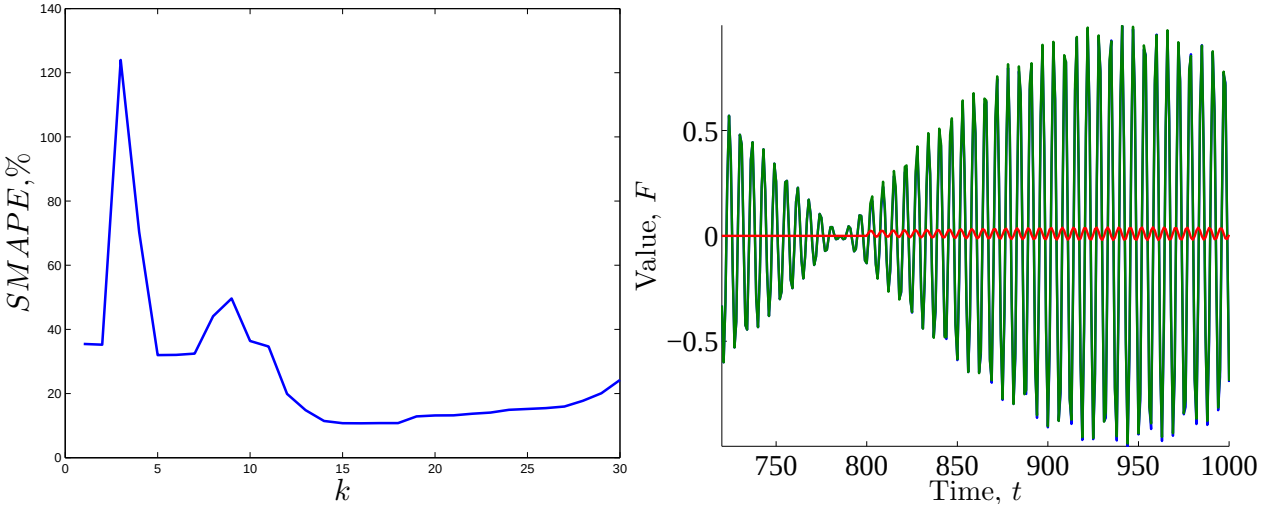


Рис. 3. Величина ошибки и построение прогноза для данных $f_1(t)$ ($k = 16$)

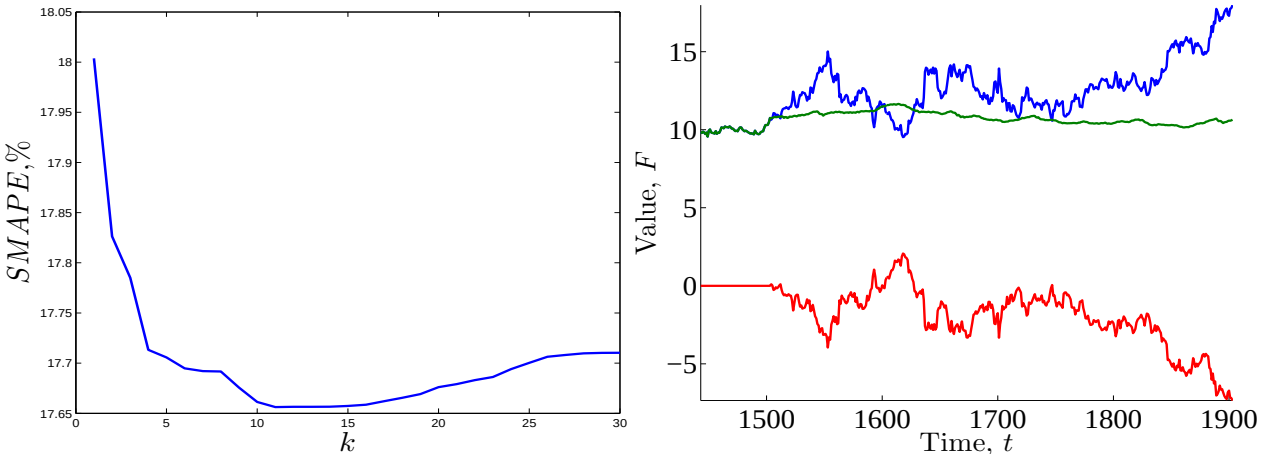


Рис. 4. Величина ошибки и построение прогноза для данных $f_2(t)$ ($k = 11$)

как степенной ряд (10). Зависимость величины ошибки SMAPE (4) от этих двух параметров и построение прогноза для рассматриваемых рядов отражены на рис.6, рис.7 и рис.8.

Метрика Минковского. Параметр метрики Минковского (3) p, очевидно, существенно влияет на качество прогноза. Также остается зависимость от количества ближайших соседей k. В данной работе рассматривается параметр метрики p в диапозоне от 1 до 10. Графики полученных зависимостей и построение прогноза приведены на рис.9, рис.10 и рис.11.

Сравнение результатов

Приведем в таблицах сравнительные результаты работы алгоритма на исследуемых данных при использовании различных метрик (1, 2, 3):

Из данных в таблицах следует, что ни одна из метрик одновременно не дает на всех исследуемых рядах оптимальный результат. Для строго периодичного модельного ряда $f_1(t)$

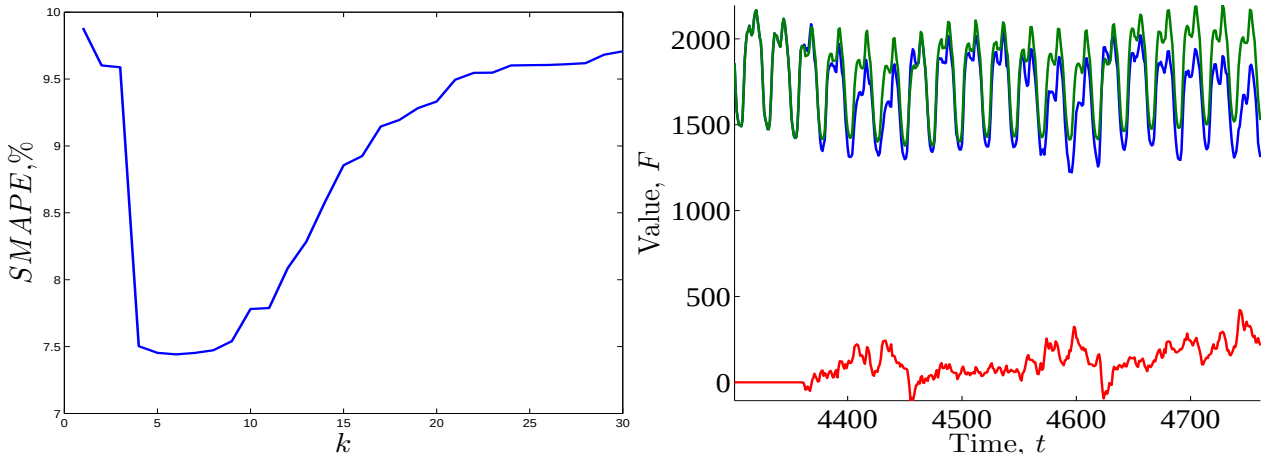


Рис. 5. Величина ошибки и построение прогноза для данных $f_3(t)$ ($k = 6$)

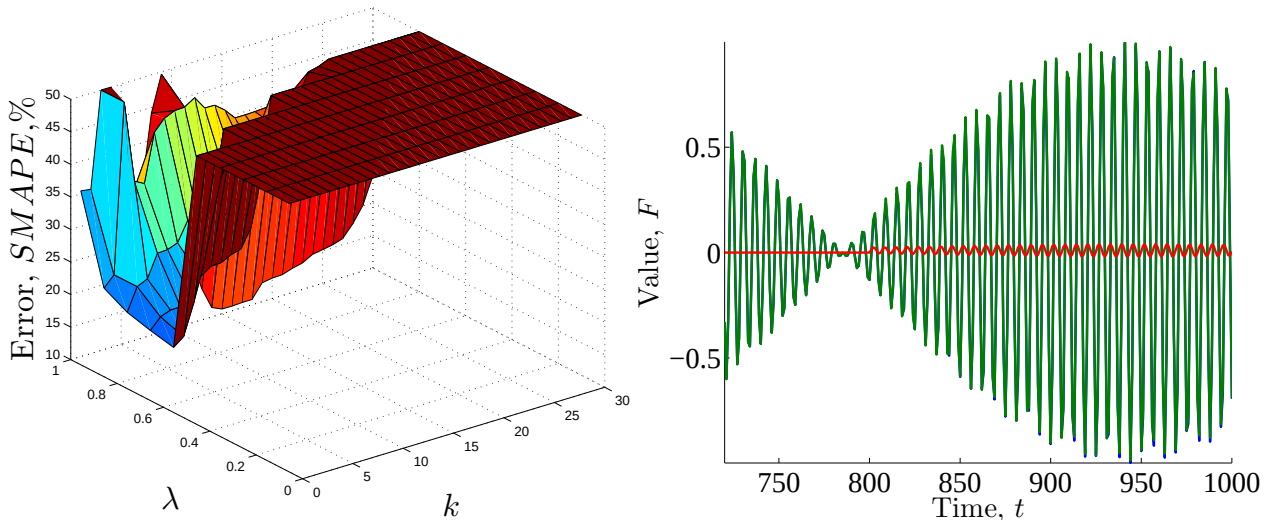


Рис. 6. Величина ошибки и построение прогноза для данных $f_1(t)$ $(k = 16, \lambda = 1)$

Metrics	(1)	(2)	(3)
best k	16	16	17
λ	1	1	1
p	2	2	4
SMAPE,%	10,71	10,71	4,91

Таблица 1. Сравнение результатов работы алгоритма на модельных данных $f_1(t)$.

оптимальной является метрика Минковского: ошибка при её использовании менее 5%. Это объясняется тем, что чем больше параметр метрики p, тем меньше весовые параметры w_i тех ближайших соседей, порядковый номер которых близок к k. На зашумленных данных о ценах на сахар $f_2(t)$ наилучший результат, как и ожидалось, дает диагонально взвешенная Евклидова метрика: в этом случае алгоритм использует намного меньшее число

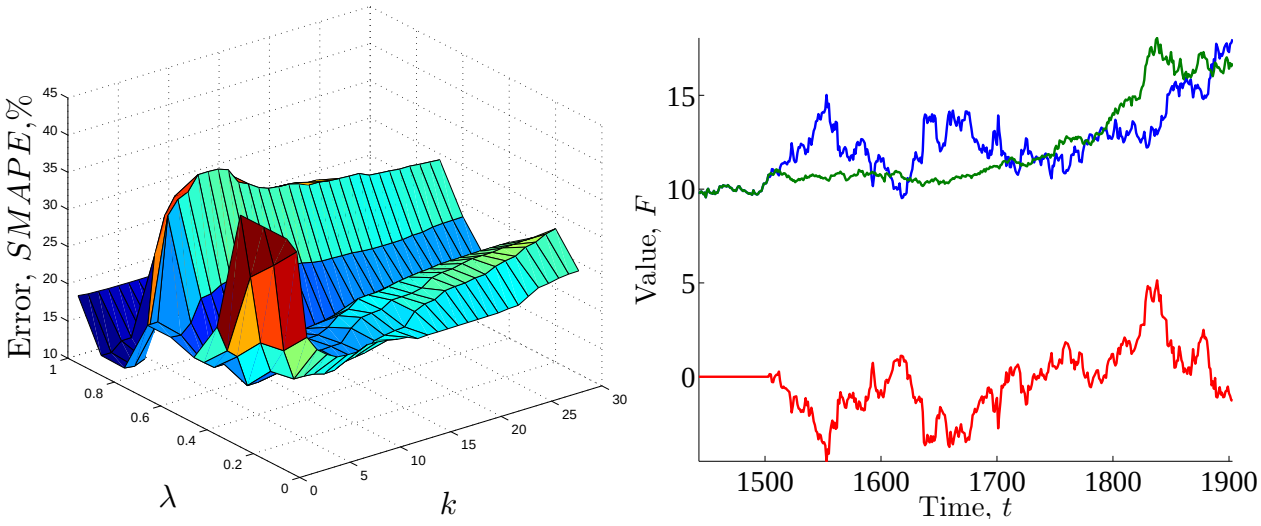


Рис. 7. Величина ошибки и построение прогноза для данных $f_2(t)$ $(k = 2, \lambda = 0.8)$

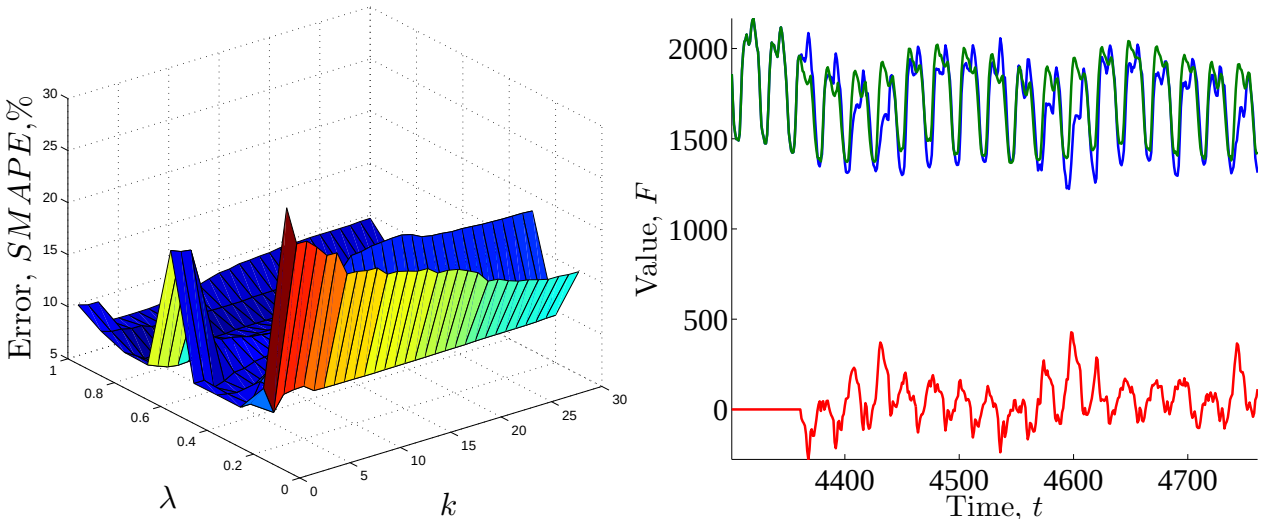


Рис. 8. Величина ошибки и построение прогноза для данных $f_3(t)$ $(k = 26, \lambda = 0,6)$

Metrics	(1)	(2)	(3)
best k	11	2	11
λ	1	0,8	1
p	2	2	2
SMAPE,%	17,66	11,3	17,66

Таблица 2. Сравнение результатов работы алгоритма на данных о ценах на сахар $f_2(t)$.

ближайших соседей, но становится менее восприимчивым к шуму за счет весового параметра λ_{ii}. На этих данных использование метрики Минковского не дает улучшения по сравнению со стандартной Евклидовой метрикой: оптимальным параметром p является $p = 2$. Для данных о потреблении электроэнергии $f_3(t)$ наилучшей также оказалась диа-

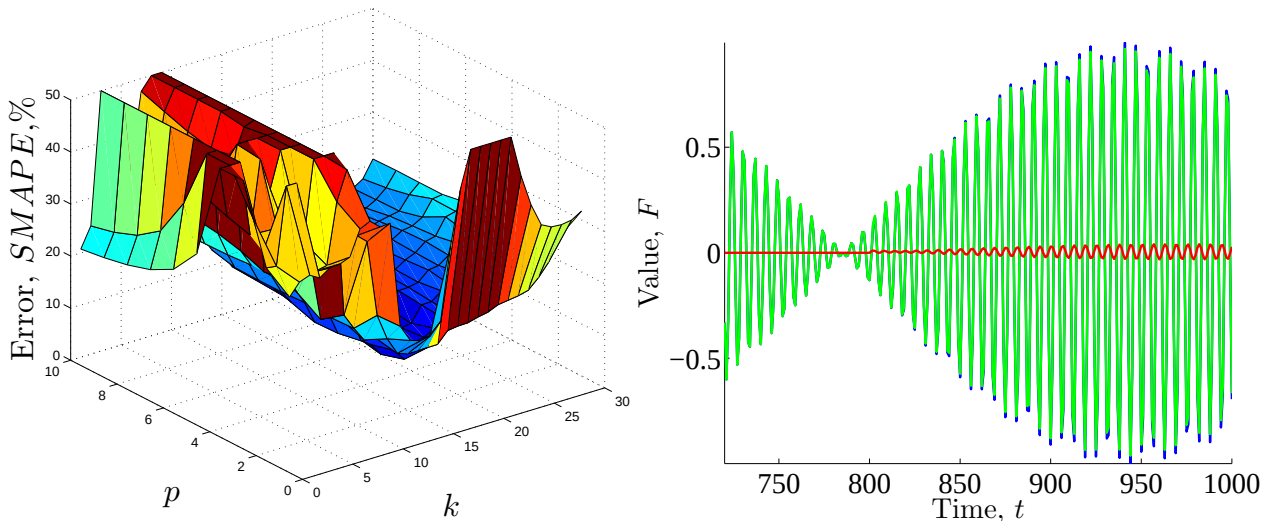


Рис. 9. Величина ошибки и построение прогноза для данных $f_1(t)$ $(k = 17, p = 4)$

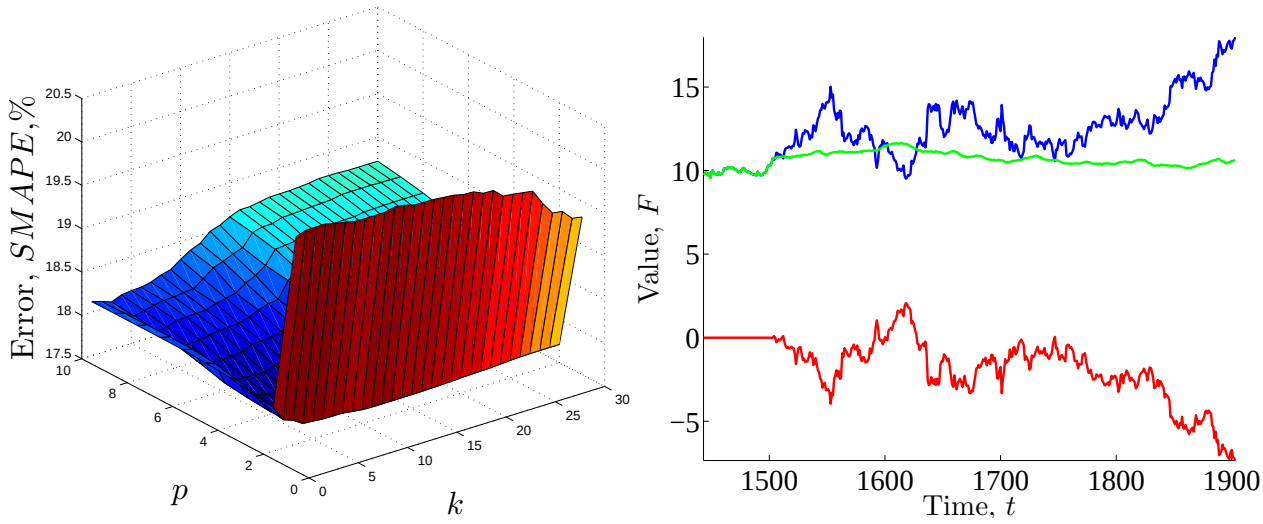


Рис. 10. Величина ошибки и построение прогноза для данных $f_2(t)(k = 11, p = 2)$

Metrics	(1)	(2)	(3)
best k	6	26	6
λ	1	0,6	1
p	2	2	5
SMAPE,%	7,44	5,74	7,37

Таблица 3. Сравнение результатов работы алгоритма на данных о потреблении электроэнергии $f_3(t)$.

гонально взвешенная Евклидова метрика, но в этом случае она, напротив, использует намного большее число ближайших соседей и меньший весовой параметр λ_{ii}, что фактически означает, что начало предыстории сильно менее важно чем ее окончание. Однако,

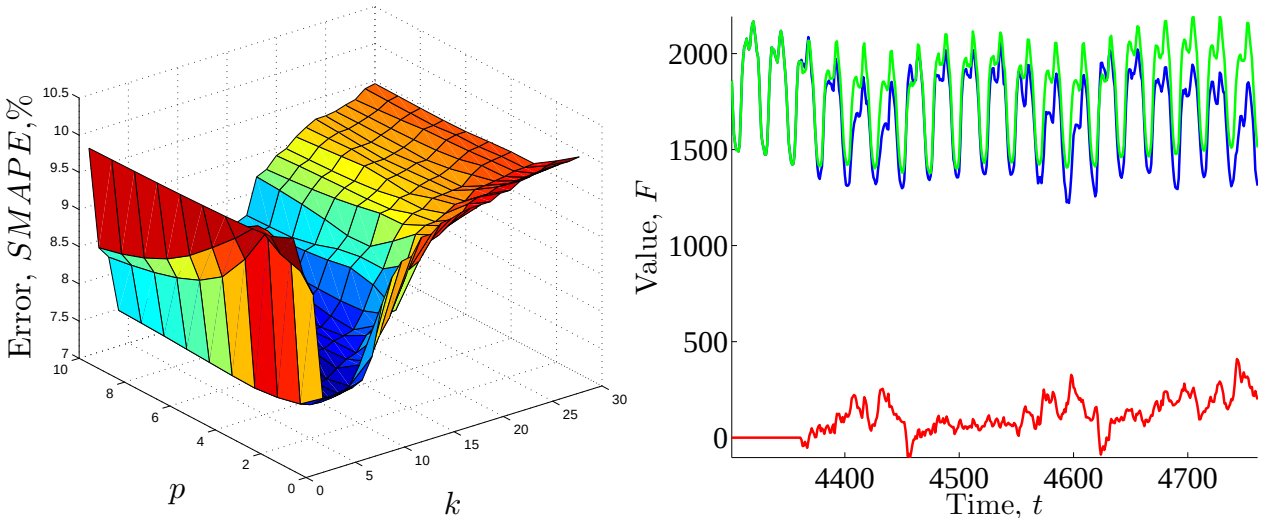


Рис. 11. Величина ошибки и построение прогноза для данных $f_3(t)$ $(k = 6, p = 5)$

разница в величине ошибки небольшая, и поэтому судить об оптимальности применения данной метрики на рядах похожего типа не представляется возможным.

Важным фактором является то, что в данной работе длина предыстории l считалась фиксированной, что ограничивает возможность судить об оптимальности применения той или иной метрики: наилучшая длина l может значительно отличаться для различных метрик.

Заключение

В данной работе рассмотрен локальный метод пронозирования временных рядов, основанный на алгоритме поиска "k ближайших соседей", исследована зависимость качества прогноза от используемой функции близости и от количества k ближайших соседей, проиллюстрированы результаты работы алгоритма на модельных рядах и реальных данных: о потреблении электроэнергии и о ценах на сахар, сравнительные результаты сведены в таблицы.

Литература

[1] McNames J.,*Innovations in local modeling for time series prediction* // Ph.D. Thesis, Stanford University, 1999.

[2] Воронцов К. В. Курс лекций *Математические методы обучения по прецедентам*

[3] Журавлев Ю. И., Рязанов В. В., и Сенько О. В. *Распознавание. Математические методы. Программная система. Практические применения.* // Фазис, Москва, 2005.

[4] Магнус Я. Р., Катышев П. К., Пересецкий А. А. *Эконометрика* // Дело, 2004, стр. 34-37

[5] Федорова В. П., *Локальные методы прогнозирования временных рядов* // Москва, 2009.

[6] Временной ряд (библиотека примеров) *http://www.machinelearning.ru/wiki/*

Оценивание вероятностей появления строк в естественном языке*

Е. А. Будников
unicorn1992@bk.ru
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

В работе рассматривается задача оценивания вероятностей появления строк в естественном языке. Для решения задачи используется модель n-грамм. Для решения проблемы большого числа параметров предлагается использовать модель n-грамм на классах. Для решения проблемы нулевых вероятностей строк предлагается использовать три дисконтные модели: Гуда-Тьюринга, Катца и абсолютного дисконтирования.
Вводятся основные определения и описываются методы, а также алгоритм построения классов в модели n-грамм на классах. Описывается проведённый эксперимент на синтетических данных.

Ключевые слова: *языковая модель, дисконтная модель, n-граммы на классах, Гуд-Тьюринг, Катц, абсолютное дисконтирование*

Введение

В задачах, связанных с распознаванием речи, часто возникает необходимость оценивать априорную вероятность появления тех или иных строк. Метод n-грамм описывается в [1, 2, 3, 4] и заключается в том, что апостеорная вероятность появления слова после некой строки зависит не от всех слов строки, а лишь от последних $n-1$.

Основными недостатками этого метода является плохая обучаемость огромного числа параметров и нулевая оценка вероятности появления на строках, которые не встречаются в процессе обучения. Для решения первой проблемы и частичного решения второй предлагается использовать метод n-грамм на классах. Он подробно описывается в [5, 1]. Этот метод заключается в том, что все слова языка разбиваются на классы, тем самым снижается число параметров, затем во время обучения настраиваются вероятности появления в языке шаблонов строк, состоящих из названий классов, а также вероятности появления слова в определённом классе.

Количество строк с нулевой вероятностью уменьшается, однако они остаются. Для перераспределения вероятностей предлагается использовать различные дисконтные модели[3, 1, 4]. В модели Гуда-Тьюринга[6] все n-граммы разбиваются на группы в зависимости от частоты появления, а затем происходит сглаживание этих частот между соседними группами. Этот метод прост в реализации, однако неустойчив. Что означает эта неустойчивость, будет пояснено ниже. Также он сглаживает и оценки вероятностей n-грамм, которые встречаются в обучении достаточно часто и могут быть признаны надёжно обученными.

В модели Катца[7] выбирается соответствующий порог, и оценки вероятностей n-грамм, частота появления которых в обучении больше этого порого, не сглаживаются. Однако эта модель также неустойчива.

Модель абсолютного дисконтирования[8] использует другой подход. Из всех ненулевых частот вычитается фиксированное число, которое потом перераспределяется между

Научные руководители: В. В. Стрижов, В. Я. Чучупал

n-граммами, не встретившимися в обучении. Можно подобрать это число так, чтобы суммарное уменьшение вероятности было таким же, как и в модели Гуда-Тьюринга.

Все эти методы были описаны в обзоре[9].

В данной работе предложены и реализованы несколько комбинаций алгоритмов оценивания вероятностей появления строк в естественном языке.

Постановка задачи

Пусть $W = \overline{w_1 w_2 \dots w_k}$ — строка из слов w_i, принадлижащих словарю Ω, которую подают на вход зашумлённого канала. Роль такого канала могут исполнять радиоэфир или человек, который переводит строку на другой язык. На выходе получим сигнал Y. По этому сигналу необходимо восстановить исходную строку. Чтобы минимизировать вероятность ошибки, необходимо взять такую строку $\hat{W}$, апостеорная вероятность которой $\Pr(\hat{W}|Y)$ максимальна:

$$\hat{W} = \arg\max_{W \in \Omega^*} \Pr(W|Y). \tag{1}$$

При фиксированном выходе Y эта задача эквивалентна максимизации совместной плотности строки W и выхода Y $\Pr(W, Y)$. Но при этом по формуле Байеса получим:

$$\Pr(W, Y) = \Pr(Y|W) \cdot \Pr(W). \tag{2}$$

Получили разбиение большой задачи на две подзадачи. Данная работа посвящена оцениванию второго множителя $\Pr(W)$.

Описание моделей

Будем обозначать подстроку строки W $w_i^j = \overline{w_i w_{i+1} \dots w_j}$, где i — позиция первого символа подстроки, а j — позиция последнего. При таких обозначениях $W \equiv w_1^k$. По формуле Байеса вероятность появления строки раскладывается в произведение апостеорных вероятностей появления каждого слова этой строки при условии извесной «предыстории», то есть подстроки, предшествующей данному слову:

$$\Pr(w_1^k) = \Pr(w_k|w_1^{k-1}) \cdot \Pr(w_{k-1}|w_1^{k-2}) \cdot \ \dots \ \cdot \Pr(w_2|w_1) \cdot \Pr(w_1) \tag{3}$$

В [9] было введено определение модели естественного языка:

Определение 1. *Моделью естественного языка назовём семейство функций*

$$f : \mathbb{R}^P \times \mathbb{R}^N \to \mathbb{R}^k,$$

где $\mathbb{R}^P$ — пространство параметров, $\mathbb{R}^N$ — пространство акустических входов, $\mathbb{R}^k$ — пространство прогнозов

Существует также и другое определение, но уже статистической модели языка[2].

Определение 2. *Статистической моделью естественного языка семейство функций*

$$f : \mathbb{R}^P \times \Omega^* \to [0, 1],$$

где $\mathbb{R}^P$ — пространство параметров, Ω^* — пространство строк, составленных из слов словаря Ω, [0, 1] — оценка вероятности появления строки в языке

Самым распространённым критерием качества модели является уровень ошибок прогнозирования. Однако измерение этого уровня требует участия систем распознавания речи. Однако можно оценивать качество модели и без их участия по тестовым строкам текста. Качество оценивается величиной *перплексии*[2].

Определение 3. *Перплексией назовём следующую величину:*

$$PP = \Pr(w_1 w_2 \dots w_k)^{-\frac{1}{k}}.$$

Перплексия является величиной, обратной к величине средней вероятности, приписываемой каждому слову тестовой строки. Модель обладает большей перплексией, если число слов, которые могут идти после заданного предыдущего, в среднем больше. Таким образом, перплексия является мерой сложности модели.

Модель n-грамм

Если не вводить никаких предположений по поводу вероятностей вида $\Pr(w_k|w_1^{k-1})$, то число параметров будет равно числу всевозможных строк языка, то есть бесконечным растущим с ростом длины строки.

В методе n-грам мы считаем две предыстории одинаковыми, если они оканчиваются на одинаковые $n-1$ слов. Другими словами,

Определение 4. *Модель естественного языка называется моделью на n-граммах, если для параметров модели выполнено условие:*

$$\Pr(w_k|w_1^{k-1}) = \Pr(w_k|w_{k-n+1}^{k-1}). \tag{4}$$

Пример. Статистическая модель биграмм задаёт следующее семейство функций:

$$f = \Pr(w_1 w_2 \dots w_n) = \Pr(w_n|w_{n-1}) \cdot \Pr(w_{n-1}|w_{n-2}) \cdot \ \dots \ \cdot \Pr(w_2|w_1) \cdot \Pr(w_1)$$

Относительно числа параметров такой модели имеет место слеследующая

Лемма 1. *Если словарь содержит V слов, то модель n-грамм содержит $V^n - 1$ параметров.*

Если словарь содержит V слов, то 1-граммы (или *униграммы*) порождают модель, имеющую $V-1$ независимых параметров: V параметров $\Pr(w_i)$ связаны равенством

$$\sum_{i=1}^{V} \Pr(\tilde{w}_i) = 1, \tag{5}$$

где $\tilde{w}_i$ — слова из словаря. 2-граммы (или *биграммы*) порождают $V^2 - 1$ независимых параметров: $V(V-1)$, имеющих форму $\Pr(w_2|w_1)$, и $V-1$, имеющих форму $\Pr(w)$. Далее по индукции легко показать, что модель n-грамм содержит $V^n - 1$ параметров.

Действительно, $V^{n-1}(V-1)$ параметров, имеющих форму $\Pr(w_n|w_1^{n-1})$, и $V^{n-1} - 1$ параметров более низкого порядка(по предположению индукции). Всего

$$V^{n-1}(V-1) + V^{n-1} - 1 = V^n - 1.$$

Настраивать параметры модели будем по тексту T.

Пусть $C(\mathbf{w})$ — число раз, которые строка $\mathbf{w}$ встретилась в обучающем тексте. Тогда в случае *униграмм* максимум правдоподобия для параметра $\Pr(w)$ достигается при $\Pr(w) = \frac{C(w)}{T}$.

Для случая n-грамм имеет место такой результат максимизации правдоподобия:

$$\Pr(w_n|w_1^{n-1}) = \frac{C(w_1^{n-1}w_n)}{\sum_w C(w_1^{n-1}w)}. \tag{6}$$

Модель n-грамм на классах

Для улучшения надёжности обучения параметров необходимо уменьшать их число, стараясь при этом не сильно потерять в точности оценок вероятностей. Также существует проблема нулевых оценок вероятностей появления строк в языке. Приведём пример. Допустим в обучающей выборке текстов многократно и в похожих ситуациях употребляются слова «мяч» и «мячик», за одним исключением: сочетания «уронила в речку мячик» и «не утонет в речке мяч» в нём присутствуют, а «уронила в речку мяч» и «не утонет в речке мячик» в нём отсутствуют. Получится, что оценка вероятностей появления соответствующих строк не только кардинально отличаются, но и пара из этих оценок и вовсе оказываются нулевыми, хотя интуитивно оценки для соответстствующих пар строк практически не должны отличаться.

Эти общие соображения естественным образом подводят нас к идее классов.

Пусть существует некоторая функция $\pi : \Omega \to G$, где Ω — множество слов, словарь, а G — множество классов слов. Тогда обозначим $Pr(w|g)$ вероятность появления в языке слова w, если известен его класс g, а $\Pr(g_n|g_1^{n-1})$ — вероятность встретить слово из класса g_n после последовательности слов, имеющих форму $g_1 g_2 \dots g_{n-1}$.

Теперь мы пожертвуем частью информации, а именно, будем настраивать лишь параметры вида $\Pr(g_n|g_1^{n-1})$ и $Pr(w|g)$.

Определение 5. *Модель n-грамм назовём моделью n-грамм на классах, если выполняется гипотеза:* $\Pr(w_k|w_1^{k-1}) = \Pr(w_k|g)\Pr(g_k|g_1^{k-1})$,*где* $k = 1, \dots, n$.

Пример. Статистическая модель биграмм на классах задаёт следующее семейство функций:

$$f = \Pr(w_1 w_2 \dots w_n) = \Pr(w_n|g_n) \cdot \Pr(g_n|g_{n-1}) \cdot \ \dots \ \cdot \Pr(w_2|g_2) \cdot \Pr(g_2|g_1) \cdot \Pr(w_1|g_1) \cdot \Pr(g_1)$$

Относительно числа параметров такой модели имеет место слеследующая

Лемма 2. *Если словарь содержит V слов и имеется C классов, то модель n-грамм на классах содержит $C^n + V - C - 1$ параметров.*

Действительно, имеется $C^n - 1$ параметров вида $\Pr(g_n|g_1^{n-1})$ (доказывается аналогично Лемме 1) и $V - C$ параметров вида $Pr(w|g)$, так всего таких вероятностей V, но для каждого класса $g \in G$ выполняется равенство:

$$\sum_{w:\pi(w)=g} \Pr(w|g) = 1. \tag{7}$$

Опишем теперь один алгоритм построения функции π на примере биграмм.

Пусть $T = (t_1, t_2, \ \dots \ , t_T)$ — обучающая выборка, причём все слова содержатся в словаре Ω. Функция правдоподобия тогда равна

$$L(T) = \Pr(T) = \prod_{x,y \in \Omega} \Pr(y|x)^{C(xy)}, \tag{8}$$

где x,y — слова из словаря, а $C(xy)$ показывает, сколько раз последовательность слов «xy» встретилась в обучающей выборке T.

Решается максимизационная задача:

$$L(T) \to \max_{\pi}. \tag{9}$$

Покажем, что имеет место

Лемма 3. *Задача максимизации 9 равносильна максимизации функции*

$$F_\pi = \sum_{g,h \in G} C(gh) \cdot \log C(gh) - 2 \sum_{h \in G} C(h) \cdot \log C(h),$$

где $C(gh)$ — функция, которая показывает, сколько раз в обучающем тексте встретились строки вида «xy», где $\pi(x) = g$, а $\pi(y) = h$

Для удобства будем использовать логарифм функции правдоподобия вместо самой функции:

$$\log L(T) = \sum_{x,y \in \Omega} C(xy) \cdot \log \Pr(y|x). \tag{10}$$

Из данного выше определения модели n-грамм на классах заключаем, что максимум правдоподобия для биграмм достигается при

$$\Pr(w_i|w_{i-1}) = \frac{C(w_i)}{C(\pi(w_i))} \cdot \frac{C(\pi(w_{i-1})\pi(w_i))}{C(\pi(w_{i-1}))}, \tag{11}$$

где $C(w_i)$ — число раз, которые слово w_i встретилось в обучающей выборке, а $C(\pi(w))$ — число раз, которые слова из класса $\pi(w)$ встретились в выборке, аналогично $C(\pi(w_x)\pi(w_y))$ — число пар вида «$\pi(w_x)\pi(w_y)$», встретившиеся в выборке.

Подставим теперь это выражение в функцию правдоподобия и преобразуем:

$$\log L(T) = \sum_{x,y \in \Omega} C(xy) \cdot \log \left(\frac{C(y)}{C(\pi(y))} \cdot \frac{C(\pi(x)\pi(y))}{C(\pi(x))} \right) \tag{12}$$

$$= \sum_{x,y \in \Omega} C(xy) \cdot \log \left(\frac{C(y)}{C(\pi(y))} \right) + \sum_{x,y \in \Omega} C(xy) \cdot \log \left(\frac{C(\pi(x)\pi(y))}{C(\pi(x))} \right)$$

$$= \sum_{y \in \Omega} C(y) \cdot \log \left(\frac{C(y)}{C(\pi(y))} \right) + \sum_{g,h \in G} C(gh) \cdot \log \left(\frac{C(gh)}{C(g)} \right)$$

$$= \sum_{y \in \Omega} C(y) \cdot \log C(y) - \sum_{y \in \Omega} C(y) \cdot \log C(\pi(y))$$

$$+ \sum_{g,h \in G} C(gh) \cdot \log C(gh) - \sum_{g,h \in G} C(gh) \cdot \log C(g)$$

$$= \sum_{y \in \Omega} C(y) \cdot \log C(y) + \sum_{g,h \in G} C(gh) \cdot \log C(gh)$$

$$-2 \sum_{h \in G} C(h) \cdot \log C(h).$$

Заметим, что первое слагаемое не зависит от выбора функции π. Поэтому его рассматривать необязательно, когда мы будем оптимизировать π.

Поэтому будем максимизировать функцию

$$F_\pi = \sum_{g,h \in G} C(gh) \cdot \log C(gh) - 2 \sum_{h \in G} C(h) \cdot \log C(h). \tag{13}$$

Приведём теперь алгоритм оптимизации функции π. Перед запуском алгоритма определяется число классов.

Имеет место следующее утверждение:

Алгоритм 1 Алгоритм построения функции π.

для всех $w \in \Omega$
  $G(w) = 1$ //инициализация
для $i = 1 \dots n$
  повторять
    для всех $c \in G$
      Переместить слово w в класс c, запомнив его предыдущий класс
      Вычислить изменения F_π для этого перемещения в c. Переместить слово w назад в его предыдущий класс
    Переместить слово w в класс, кторый больше всего увеличивает F_π, или никуда не перемещать, если увеличения ни на каком перемещении не происходит
  пока s

Лемма 4. *Алгоритм 1 сходится к локальному минимуму F_π.*

Утверждение очевидно и следует из того, что на каждом шаге значение F_π увеличивается.

Дисконтная модель

Рассмотрим событие S, которое встретилось s раз, а общее количество наблюдений A. Тогда оценка вероятности S по принципу наибольшего правдоподобия будет равна

$$\Pr(S) = \frac{s}{A}. \tag{14}$$

Но тогда, в соответствии с этим принципом, событиям, которые не были встречены среди обучающего текста T, будут приписаны нулевые вероятности, а значит, будучи встреченными на тесте, они никогда не будут распознаны.

Чтобы справиться с этой проблемой, можно поступить следующим способом. В оценке вероятности события вместо числа s брать

$$s' = d_s \cdot s, \tag{15}$$

где d_s — множитель, зависящий от числа раз, которые событие встретилось в обучающем тексте. Тогда получим дисконтную оценку вероятности события S:

$$\Pr{}_{discount}(S) = \frac{s'}{A} = \frac{d_s \cdot s}{A}. \tag{16}$$

Различные дисконтные методы различаются стратегией выбора d_s.

Обозначим c_s число всех событий которые встретились в процессе обучения ровно s раз. Тогда общее число наблюдений $A = \sum_{s \geqslant 1} c_s \cdot s$. Получается, что таким образом мы перераспределили оценки вероятности между событиями и оставили на все не встретившиеся в обучении слова $1 - \frac{1}{A} \sum_{s \geqslant 1} d_s \cdot c_s \cdot s$. Если c_0 — число таких событий, то оценка вероятности каждого из них равна

$$\frac{1}{c_0} \left(1 - \frac{1}{A} \sum_{s \geqslant 1} d_s \cdot c_s \cdot s \right). \tag{17}$$

Дисконтная модель Гуда-Тьюринга

В статье[6] предлагается следующая стратегия выбора множителя:

$$d_s = (s+1)\frac{c_{s+1}}{s \cdot c_s}. \tag{18}$$

Эта стратегия называется оценкой Гуда-Тьюринга. Несмотря на очевидную простоту стратегии, у неё есть существенный недостаток: она проваливается в случае, если $c_a = 0$ для некоторого a и существует $b > a$, такой, что $c_b \neq 0$. Также существенно, что дисконтирование необходимо для параметров, оценка которых является ненадёжной, то есть для тех событий, которые встречаются в обучении менее некоторого количества раз k, выбранного априори.

Дисконтная модель Катца

Решение этой проблемы было предложено в [7]. Пусть есть некое, достаточно большое число k, такое что все оценки вероятностей событий, встретившихся в процессе обучения более k раз, признаем надёжными. При этом d_s будет выглядеть так:

$$d_s = \begin{cases} \frac{(s+1)\frac{c_{s+1}}{s \cdot c_s} - (k+1)\frac{c_{k+1}}{c_1}}{1-(k+1)\frac{c_{k+1}}{c_1}} & , \ 1 \leqslant s \leqslant k \\ 1 & , \ s > k \end{cases} \tag{19}$$

Этот метод тоже нестабильный, так как возможны ситуации, когда $d_s < 0$.

Модель абсолютного уменьшения

Одной из альтернатив модели Гуда-Тьюринга является модель абсолютного уменьшения[8]. В этой модели происходит уменьшение числа a для каждого события на фиксированное число m.

$$d_s = \frac{s-m}{s}. \tag{20}$$

Для того чтобы уменьшение суммарной вероятности было таким же, как в модели Гуда-Тьюринга, необходимо, чтобы

$$m = \frac{c_1}{\sum_{s \geqslant 1} c_s}. \tag{21}$$

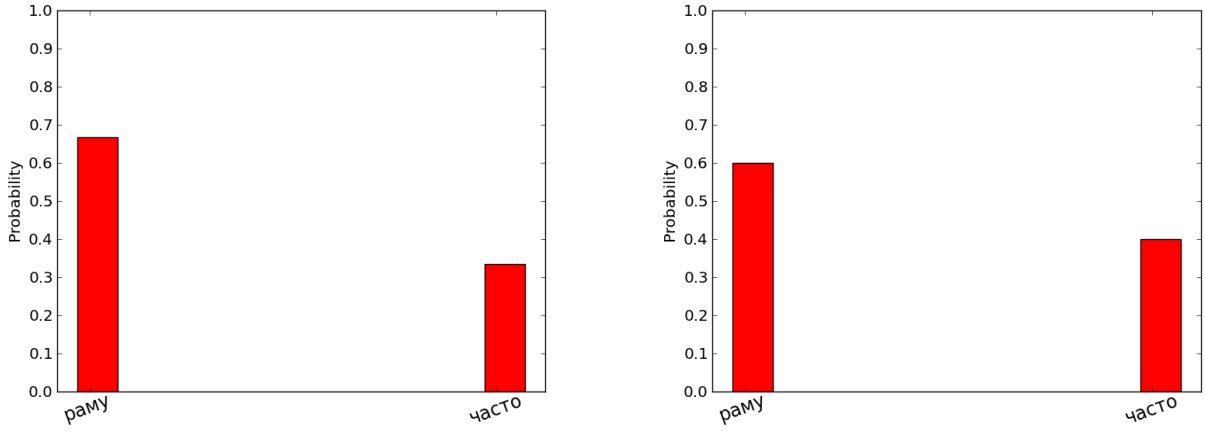
Вычислительный эксперимент

Целью вычислительного эксперимента являлась демонстрация работы комбинаций алгоритмов на небольшом массиве синтетических данных, состоящих из небольшого текста на тему «Мама моет раму». В первой серии экспериментов оценивалось распределение вероятностей появления слова после заданной строки текста. Во второй серии экспериментов оценивалась перплексия различных тестовых строк: встречающейся в обучающем тексте и двух не встречающихся в тексте.

В обеих сериях для методов n-грамм и n-грамм на классах проводилось по четыре типа экспериментов: без дисконтирования и по каждому из трёх типов дисконтирования.

Оценивание распределения вероятностей после заданной фразы

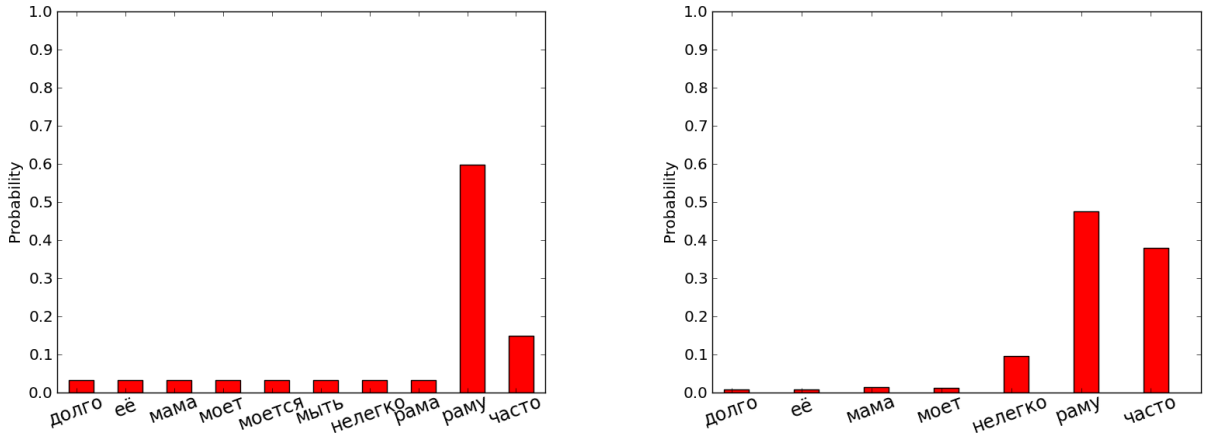
В первой серии экспериментов оценивалось распределение вероятностей появления слова после фразы «Мама моет...»



Метод n-грамм Метод n-грамм на классах

Рис. 1. Методы без дисконтирования

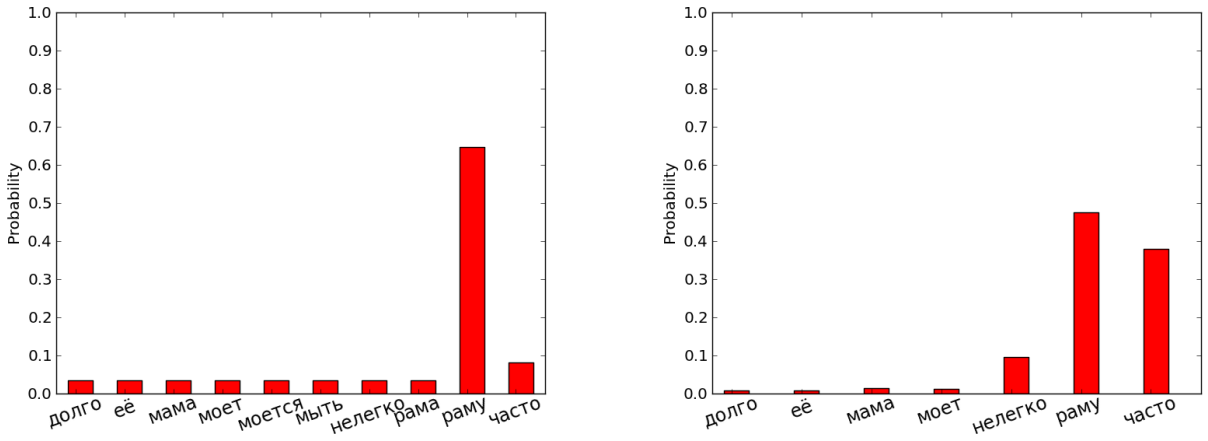
Оба метода без дисконтирования сработали примерно одинаково, распределив лишь немного иначе вероятности между двумя вариантами продолжения. Также читатель может заметить, что метод n-грамм на классах немного сгладил разницу между вероятностями. Это связано с тем, что алгоритм 1 определил слова «раму» и «часто» в один класс, а вероятности между этими словами распределяются в зависимости от суммарной частоты появления в обучающем тексте, а не только после строки «Мама моет...», а точнее, шаблона строки «$g_1 g_2$...», где g_1 — класс слова «мама», а g_2 — класс слова «моет».



Метод n-грамм Метод n-грамм на классах

Рис. 2. Модель дисконтирования Гуда-Тьюринга

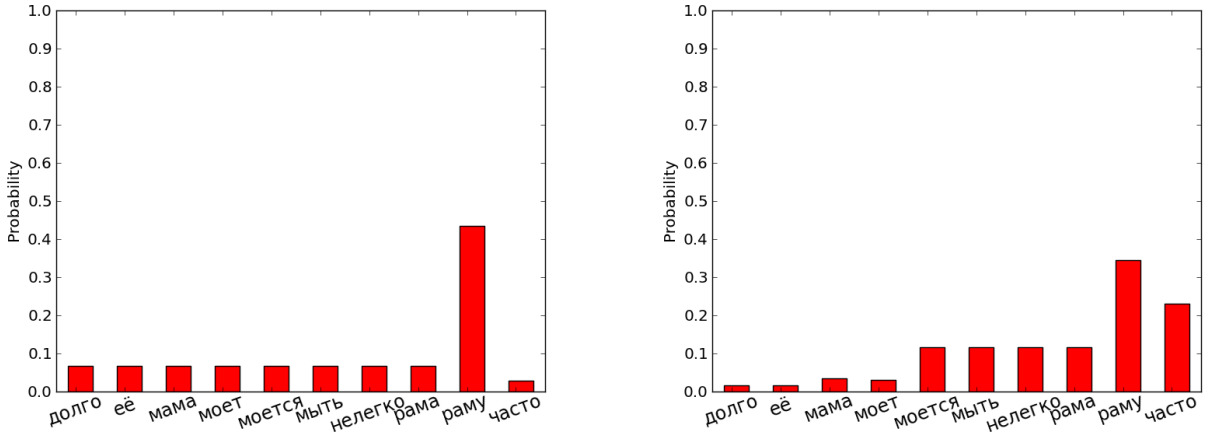
На рисунке 2 продемонстрирован метод дисконтирования Гуда-Тьюринга. В графики были включены лишь варианты с вероятностями > 0.004. Читатель может обратить внимание на снизившуюся оценку вероятности слова «часто» в методе n-грамм. Это связано с тем, что метод дисконтирования предполагает, что оценка вероятности появления события, встретившегося однажды или дважды в процессе обучения, не должна существенно отличаться от оценки вероятности появления события, в процессе обучения не встретившегося.



Метод n-грамм Метод n-грамм на классах

Рис. 3. Модель дисконтирования Катца

На рисунке 3 можно заметить, что в модели дисконтирования Катца надёжно обученные параметры не сглаживаются.



Метод n-грамм Метод n-грамм на классах

Рис. 4. Модель асболютного дисконтирования

В методе абсолютного дисконтирования при использовании метода n-грамм происходит парадоксальная ситуация: оценка вероятности события, встречавшегося в обучении, в итоге оказывается ниже оценки вероятности события, которое в обучении не встретилось.

Это объясняется высокой долей биграмм, которые встретились в обучении только один раз, среди всех встретившихся в обучении биграмм.

Оценка сложности модели на тестовых строках

Во второй серии экспериментов оценивалась перплексия различных тестовых строк.

Таблица 1. Перплексия подстроки из обучающего текста «Мама моет часто».

Модель дисконтирования	n-граммы	n-граммы на классах
Без дисконтирования	2.5	2.06186
Гуд-Тьюринг	5.62562	3.82051
Катц	9.66017	1.9987
Абсолютное	3.00793	2.71695

Таблица 2. Перплексия подстроки «Мама моет долго», которая не встречается в обучающем тексте.

Модель дисконтирования	n-граммы	n-граммы на классах
Без дисконтирования	∞	6.12372
Гуд-Тьюринг	12.2359	25.5717
Катц	14.8572	13.3778
Абсолютное	8.53946	18.0222

Читатель может заметить по таблицам 1 и 2, что самым предпочтительным пока выглядят метод n-грамм на классах без дисконтирования и метод n-грамм на классах с дисконтированием Катца. Они надёжно оценивают вероятности строк и обладают минимальными перплексиями.

Однако давайте посмотрим на оценку перплексии ещё одной строки.

Таблица 3. Перплексия подстроки «Долго её моет».

Модель дисконтирования	n-граммы	n-граммы на классах
Без дисконтирования	∞	∞
Гуд-Тьюринг	6.53827	6.14817
Катц	6.84374	8.76256
Абсолютное	8.12289	7.77689

В таблице 3 читатель может заметить, что если метод n-грамм на классах без дисконтирования даёт нулевую оценку вероятности строки, то более предпочтительными являются методы с дисконтированием Гуда-Тьюринга или абсолютным дисконтированием.

Заключение

В работе были рассмотрены методы оценивания вероятностей появления строк в языке, основанные на n-граммах. Каждый из рассмотренных методов обладает, как показал вычислительный эксперимент, своими достоинствами и недостатками. К достоинствам метода n-грамм без дисконтирования можно отнести линейную по размеру обучающего текста сложность алгоритма настройки параметров, к недостаткам — большое число параметров и, как следствие, плохую их обучаемость, а также нулевую оценку вероятности появления в языке n-грамм, которые не встретились в процессе обучения.

К достоинствам метода n-грамм на классах можно отнести, что число параметров линейно по размеру словаря и квадратично по числу классов, локальную оптимальность решения задачи разбиения слов на классы. Недостатками являются высокая вычисли-

тельная сложность алгоритма, а также наличие нулевых оценок вероятностей, хоть и на меньшем количестве строк с сравнении с методом n-грамм.

Дисконтные модели решают проблему нулевых оценок вероятностей появления строки в естественном языке, однако они могут работать неадекватно, если велика доля ненадёжно обученных параметров. Также недостатком моделей Гуда-Тьюринга и Катца является их неустойчивость.

Литература

[1] Xuedong Huang, Alex Acero, and Hsiao-Wuen Hon. *Spoken Language Processing, A Guide to Theory, Algorithm, and System Development.* Prentice Hall PTR, 2001.

[2] Frederick Jelinek. *Statistical Methods for Speech Recognition.* The MIT Press, Cambridge, Massachusetts, 1997.

[3] Yoshihiko Gotoh and Steve Renals. Statistical language modelling. In Steve Renals and Gregory Grefenstette, editors, *ELSNET Summer School*, volume 2705 of *Lecture Notes in Computer Science*, pages 78–105. Springer, 2000.

[4] Steve Young and Gerrit Bloothooft, editors. *Corpus-Based Methods in Language and Speech Processing.* Kluwer Academic Publishers, Dordrecht, 1997.

[5] Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, and Robert L. Mercer. Class-based n-gram models of natural language. In *Proceedings of the IBM Natural Language ITL*, pages 283–298, Paris, France, March 1990.

[6] I. J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3 and 4):237–264, 1953.

[7] Slava M. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3):400–401, March 1987.

[8] Hermann Ney, Ute Essen, and Reinhard Kneser. On structuring probabilistic dependencies in stochastic language modelling. *Computer Speech and Language*, 8:1–38, 1994.

[9] Егор Алексеевич Будников. Обзор некоторых статистических моделей естественных языков. *Машинное обучение и анализ данных*, 1:245–250, декабрь 2011.