Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего профессионального образования
«Московский физико-технический институт (государственный университет)»

Факультет управления и прикладной математики

Кафедра проблем передачи информации и анализа данных

На правах рукописи

УДК _____

Трофимов Михаил Игоревич

Исследование стратегий активного обучения

Выпускная квалификационная работа бакалавра

Направление подготовки: 010900 Прикладные математика и физика

Заведующий кафедрой	_____	/ Кулешов А.П. /
Научный руководитель	_____	/ Карпенко С.М. /
Студент	_____	/ Трофимов М.И. /

г. Москва
2014

Содержание

Введение

Классическая постановка задачи машинного обучения с учителем опирается на существование обучающей выборки. На практике это означает, что необходимо заранее подготовить и разметить объекты. В идеальном случае обучающий набор должен быть репрезентативен относительно генеральной совокупности. Разметка часто является самой дорогой операцией в обучении, так как требует существенных временных затрат или привлечения специалистов в предметной области задачи.

Каждый пример может предоставлять различное количество информации. Идея заключается в отборе объектов, предоставляющих наибольшее количество информации и имеющих бóльшее значение для процесса обучения. Однако выделение сложных обучающих примеров, предоставляющих максимум информации для обучения, является трудной задачей для оператора. Активное обучение является инструментом для выделения таких объектов.

В литературе ([1], [2]) описаны 3 основных сценария использования активного обучения. Первый из них, известный как «синтез запросов» (англ: query synthesis) подразумевает, что алгоритм может синтезировать объект для запроса его метки. Однако часто в пространстве признаков точки, соотвествующие реальным объектам, лежат на некотором многообразии. Запрос, синтезируемый алгоритмом, часто лежит вне этого многообразия, т.е. соответствующего реального объекта может просто не существовать. Такой подход редко используется в практических приложениях.

Следующий сценарий носит название «потоковый» (англ: stream-based) и предполагает, что на каждом шаге на вход поступает одна точка, и алгоритм должен единоразово принять решение о том, запрашивать ли метку или игнорировать её. Проигнорированный объект в дальнейшем не используется в обучении. В основном этот подход применяется в ситуациях, когда имеется непрерывный поток данных с датчиков или сенсоров и ограниченый ресурс для хранения

этих данных.

Помимо уже упомянутых, рассматривается сценарий «с фиксированным пулом» (англ: pool-based). Он предполагает, что алгоритм может запрашивать точки только из некоторого ранее зафиксированного множества. Этот подход наиболее близок к классической постановке задачи классификации и чаще всего применяется на практике.

Методы активного обучения можно разделить на категории по идеям, лежащим в основе, и по требованиям, предъявляемым к базовому классификатору. В литературе([1], [2], [3]) рассматриваются 5 основных подходов к решению данной задачи.

Один из подходов основан на близости объектов к границе, разделяющей классы. Это эквавалентно тому, что если базовый классификатор умеет оценивать вероятность принадлежности объекта к каждому из классов, то можно выбрать те примеры, которые были классифицированы неуверенно. Если их разметить, добавить к обучающему множеству и настроить модель на обновленной выборке, то можно ожидать качественное улучшение классификатора. Эта идея носит название «стратегия неуверенности» (англ: uncertainty sampling). Такой подход является эвристическим и не имеет строгого теоретического обоснования, однако, в силу своей простоты и эффективности, широко используется на практике и часто упоминается в научных работах.

Другой подход основан на поиске в пространстве гипотез. Его идея заключается в том, чтобы выбирать те точки, которые как можно сильнее сужают множество гипотез, согласующихся с имеющейся размеченной выборкой. Такая стратегия носит название «выбор по рассогласованности» (англ: query by disagreement, [4], [5]), для нее есть некоторые теоретические результаты, в том числе получены оценки числа примеров, необходимых для достижения определенного уровня качества ([6]). Этот подход широко используется на практике ([7], [2], [8]) и является предметом научных исследований в настоящее время.

Следующая стратегия основана на идее о прямой минимизации ожидаемой

ошибки классификатора после добавления новой точки в обучающую выборку ([1]). От базового классификатора требуется оценка вероятностного распределения меток объекта. Делается предположение о том, что истинное распределение меток может быть аппроксимировано оценкой классификатора. Это дает возможность вычислять математическое ожидание ошибки классификатора и выбирать для размерки тот пример, который эту величину минимизирует. Подобные методы требуют многократного обучения базового классификатора, а потому работают только на малых выборках.

Кроме того, в литературе рассматривается стратегия, основанная на минимизации дисперсии параметров модели классификатора. Этот подход очень тесно связан с задачей планирования эксперимента в статистике, имеет хорошее теоретическое обоснование, но накладывает сильные ограничения на используемый базовый классификатор.

Помимо упомянутых, существуют еще методы, которые используют только информацию о структуре данных, без учета модели. В работе [9] описано использование иерархической кластеризации и последовательный спуск от более крупных единиц к более мелким с запросом меток. Другая реализация этой же идеи заключается в построении графа данных с распространением меток по этому графу. Описание этого метода можно найти в работе [10]. Кроме того, кластеризация может быть использована как дополнение к приведенным выше стратегиям, подробнее об этом можно прочитать в работе [1].

В данной работе рассматривается задача активного обучения классификатора для непересекающихся классов с заранее фиксированным пулом точек и исследуются стратегии, опирающиеся на неуверенность и рассогласованность. Такой выбор обусловлен разумной вычислительной сложностью и широкой применимостью данных методов.

Глава 1

Постановка задачи

1.1. Используемые обозначения

$\mathcal{D} = \{d_1, d_2, \ldots, d_k\}$ – множество классов

$\mathcal{S}^k \subset \mathbb{R}^k$ – вероятностный k-мерный симплекс

$x \in \mathbb{R}^m$ – объект признакового пространства

$y \in \mathcal{D}$ – класс, метка объекта

$\mathcal{U} = \{x_n\} \subset \mathbb{R}^m$ – множество неразмеченных примеров, пул

$\mathcal{L} = \{(x_n, y_n)\} \subset \mathbb{R}^m \times \mathcal{D}$ – множество размеченных примеров

$\mathcal{H}$ – множество базовых классификаторов

$h_{\mathcal{L}}(x) : \mathbb{R}^m \to \mathcal{D}$ – классификатор, построенный по обучающей выборке $\mathcal{L}$

$prob_h(x) : \mathbb{R}^m \to \mathcal{S}^k$ – оценка классификатора h распределения вероятности принадлежности x к каждому из классов

$\phi_h(x) : \mathbb{R}^m \to \mathbb{R}$ – функция критерия

$label(x) : \mathbb{R}^m \to \mathcal{D}$ – функция оракула

$d(x, h_1, \ldots, h_q) : \mathbb{R}^m \times \mathcal{H}^q \to \mathbb{R}$ – функция рассогласованности

$acc(h) : \mathcal{H} \to \mathbb{R}$ – используемый функционал качества классификатора

1.2. Математическая постановка задачи

Рассматривается сценарий с фиксированным пулом доступных точек.

Пусть дано множество неразмеченных примеров $\mathcal{U} \subset \mathbb{R}^m$ и множество размеченных примеров $\mathcal{L} \subset \mathbb{R}^m \times \mathcal{D}$. Потребуем, чтобы все классы были представлены[1] в множестве $\mathcal{L}$. Этому требованию соотвествует следующее условие:

[1] Случай, когда не все классы представлены в стартовом множестве, носит название «active discovery» и его исследованию посвящены работы [11], [12]

$$\forall d_i \in \mathcal{D} \quad \exists (x_j, y_j) \in \mathcal{L} \quad \Rightarrow \quad y_j = d_i$$

На каждом шаге алгоритм активного обучения должен выбрать одну точку из пула и запростить для нее метку. В итоге требуется построить последовательность $\mathcal{Z} = \{x_i, label(x_i)\}_{i=1}^{C}$ такую, что:

$$\{\mathcal{Z}\} = \arg\max_{\mathcal{Z}^* \subset \mathcal{U}} acc(h_{\mathcal{L} \cup \mathcal{Z}^*}) \tag{1.1}$$

$$|\mathcal{Z}^*| \leq C$$

где C - заранее заданная константа, имеющая смысл бюджета.

Заменим задачу оптимизации по подмножеству жадной стратегией поэлементного добавления. Тогда постановка следующая:

найти такой $x^* \in \mathcal{U}$, что

$$x^* = \arg\max_{x \in \mathcal{U}} (acc(h_{\mathcal{L} \cup \{x, label(x)\}}) - acc(h_{\mathcal{L}})) \tag{1.2}$$

Отметим, что x^* не может быть найден прямым перебором, т.к. значение $label(x)$ заранее не известно.

1.3. Общий алгоритм

На поиск прямого решением поставленной выше локальной задачи направлены методы, основанные на минимизации ожидаемой ошибки. Они используют сильные вероятностные предположения и без сильных ограничений на базовый классификатор являются вычислительно неприемлимыми. В общем случае решение сформулированной выше локальной задачи не известно, и потому используются эвристические методы, основанные на подмене аргумента оператора $\arg\max$ в условии 1.2 на функцию полезности обучающего примера (она же - функция критерия) $\phi_h(x)$. Когда $\phi_h(x)$ определена, общий алгоритм активного обучения может быть представлен в виде листинга 1.

Исходные параметры: множества $\mathcal{U}$ и $\mathcal{L}$

Результат: множество $\mathcal{L}^*$

$i = 0$;

до тех пор, пока $i < C$ **выполнять**

  $h(x) \leftarrow h_{\mathcal{L}}(x)$;

  $x = \arg\max_{x \in \mathcal{U}} \phi_h(x)$;

  $y = label(x)$;

  $\mathcal{L} \leftarrow \mathcal{L} \cup \{(x, y)\}$;

  $\mathcal{U} \leftarrow \mathcal{U} \setminus \{x\}$;

  $i = i + 1$;

конец цикла

Алгоритм 1: Общий алгоритм активного обучения

Уже упомянутые методы, основанные на минимизации ожидаемой ошибки, укладываются в этот алгоритм, если в качестве $\phi_h(x)$ взять мат.ожидание ошибки классификации при добавлении примера x, умноженное на -1. Общая идея заключается в построении эффективно вычислимой функции критерия такой, что

$$acc(h_{\mathcal{L}\cup\{x^+,label(x^+)\}}) \text{ близко к } \max_{x\in\mathcal{U}} acc(h_{\mathcal{L}\cup\{x,label(x)\}}),$$

где $x^+ = \arg\max_{x\in\mathcal{U}} \phi_h(x)$.

Стартовой точкой для сравнения различных стратегий активного обучения является стратегия случайного выбора. Она подразумевает, что точки из пула выбираются случайным образом с одинаковыми вероятностями $\frac{1}{|\mathcal{U}|}$. Этого можно добиться, задав функцию критерия как случайную величину

$$\phi_h^{random}(x) \sim \mathcal{N}(0, 1)$$

Определенная таким образом функция $\phi_h(x)$ укладывается в рамки приведенного общего алгоритма.

Глава 2

Стратегии, основанные на неуверенности

Стратегии этого типа опираются на идею о том, что трудными для классификатора и имеющими наибольшее влияние на процесс обучения являются примеры, лежащие близко к границе, раделяющей классы. Эвристика заключается в формализации понятия неуверенности и допускает различные варианты определения функции критерия.

2.1. Случай бинарной классификации

Положим $k = 2$, т.е. $\mathcal{D} = \{0, 1\}$. Так же предполагаем, что базовый классификатор может возвращать свою оценку вероятности принадлежности точки к определенному классу, т.е. определена функция $prob_h(x)$. Тогда функция критерия принимает вид:

$$\phi_h(x) = \min\{prob_h(x)\} = \min\{\hat{\Pr}y_0, \hat{\Pr}y_1\}$$

2.2. Многоклассовый случай

Рассмотрим случай $k \geq 2$, т.е. $\mathcal{D} = \{d_1, d_2, \ldots, d_k\}$ и предположим, что определена функция $prob_h(x)$. Тогда существуют различные формализации понятия близости к границе, разделяющей классы. ([1])

Рассмотрим следующую формулировку - «близость точки к границе определяется расстоянием до ближайшей из границ». В литературе такая стратегия носит название «минимаксная» (англ: minmax, least confident), функция критерия принимает вид:

$$\phi_h^{least\,confident}(x) = 1 - \hat{\Pr}\{h(x) = d_i\},$$

где $d_i = \arg\max_{prob_h(x)}$

Один из способов интерпретации этой функции - ожидаемые 0\1 потери классификатора. Важно отметить, что такая стратегия опирается только на самый вероятный класс по оценке базового классификатора, то есть только на пик вероятностного распределения, игнорируя все остальное. Один из способов учесть больше информации - использовать разницу между самым вероятным и вторым по вероятности классом. Такой подход носит название «стратегия зазора» (англ: margin sampling), и алгоритм для вычисления ее функции критерия имеет вид:

Исходные параметры: классификатор h, пример x

Результат: значения функции критерия

1 $probs \leftarrow prob_h(x)$;

2 отсортировать $probs$ по убыванию;

3 $result \leftarrow probs(1) - probs(2)$;

4 вернуть $-result$

Алгоритм 2: Алгоритм вычисления ϕ^{margin}

Чтобы соотвествовать общему алгоритму, в котором решается задача максимизации, на последнем шаге алгоритма результат умножается на -1. Интуитивно понятен смысл этой стратегии - пример тем полезнее, чем меньше зазор между двумя наиболее вероятными классами.

Другая эвристика предлагает в качестве меры неопределенности использовать всю информацию о распределении меток, а именно - оценивать энтропию ответа базового классификатора. Функция критерия в таком случае будет иметь вид:

$$\phi_h^{entropy}(x) = -\sum_{i=1}^{k} \hat{\Pr} y_i \times \log \hat{\Pr} y_i$$

Отметим, что в случае бинарной классификации все эти обобщения вырождаются в уже упомянутую стратегию выбора точки, ближайшей к границе, разделяющей классы.

Все три приведенные стратегии обобщают одну и ту же исходную идею,

но имеют разные свойства. Ниже представлены визуализации в модельном случае: в вершинах треугольника вероятности принадлежности к классу близки к единице, а для любой точки плоскости вероятность принадлежности к классу обратно пропорциональна расстоянию до соответсвующего угла.

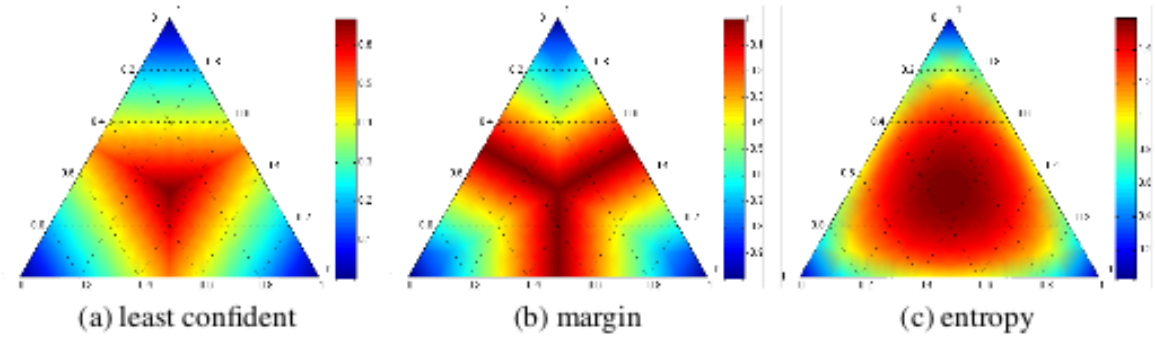

Рис. 2.1. Визуализация функций критерия

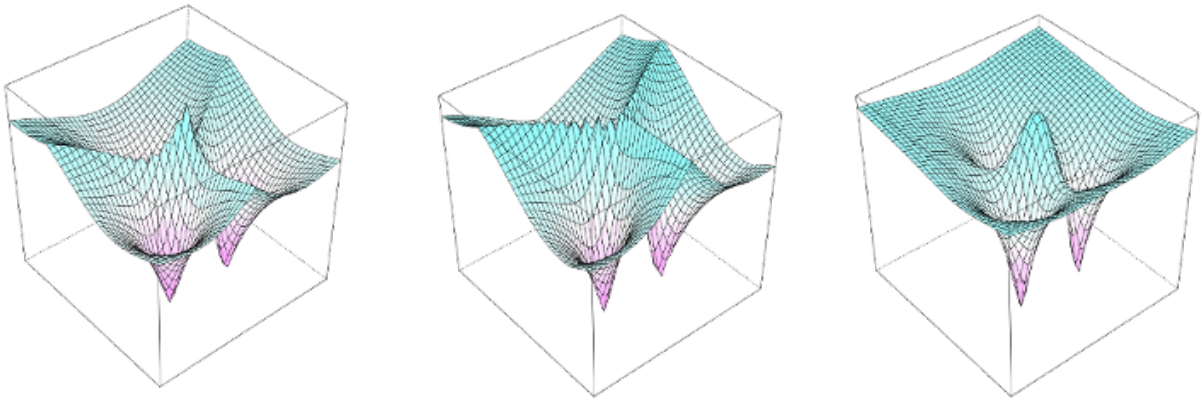

Рис. 2.2. Трехмерная визуализация функций критериев minmax, margin и entropy соответственно

Во всех случаях самой неуверенной будет признана точка, находящая на границе всех трех классов, а точки, соотвествующие углам вероятностного симплекса, будут охарактеризованы как наиболее уверенные. Разница проявляется на остальной части пространства. Например, энтропийный критерий не выделяет точки, у которых неуверенность высока только по одной паре классов. В

свою очередь, критерий зазора и минимаксный выделяют именно точки, которые имеют близкие вероятности по паре классов.

2.3. Численный эксперимент

Экспериментальное сравнение упомянутых эвристических критериев проводилось на различных базах данных, но методика была постоянна. Данные разделялись на стартовое множество $\mathcal{L}$, множество пула неразмеченных точек $\mathcal{U}$ и тестовое множество $\mathcal{T}$, которое использовалось только для оценки качества классификатора. Функцию $acc(h)$ определили как имперический риск на множестве $\mathcal{T}$, т.е.

$$acc(h) = \frac{1}{|\mathcal{T}|} \sum_{(x,y)\in\mathcal{T}} [h(x) = y]$$

На каждой итерации обучалась базовая модель $h(x)$, на ее основе вычислялись значения функции критерия $\phi_h(x)$ для всех $x \in \mathcal{U}$. Использовался пакетный режим (англ: batch mode, подробнее в [1]), то есть на каждом шаге фиксированное количество точек с наибольшими значениями функции критерия добавлялись в обучающую выборку. После каждой итерации производилась оценка качества модели.

В качестве базового классификатора использовалась модель «случайный лес»[13]. Такой выбор обусловлен простотой настройки модели, возможностью параллельного обучения, малым числом гиперпараметров и способностью работать как с вещественными, так и с категориальными признаками[14].

2.3.1. Модельный пример

Данные были сгенерированны согласно смеси четырех гауссиан, распределение которое представлено на рис 2.3.

Параметры эксперимента: $|\mathcal{L}| = 10$, $|\mathcal{U}| = 170$.

Пример начальной инициализации показан на рис 2.4.

Результаты эксперимента приведены на рис 2.5.

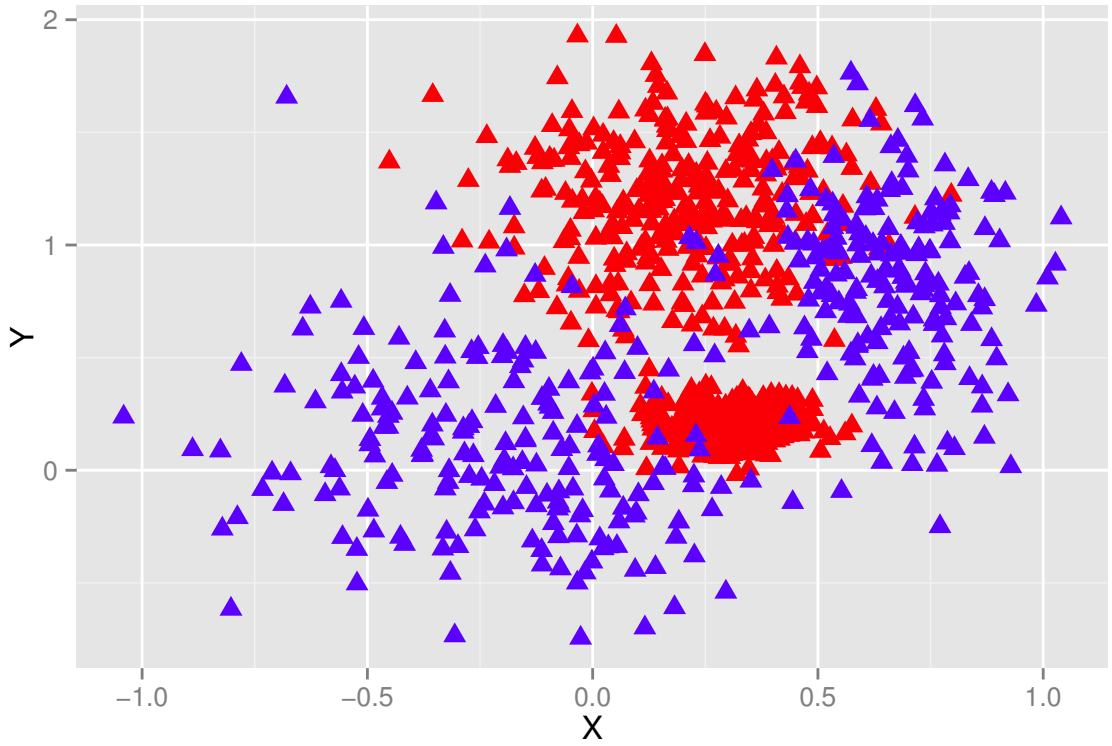

Рис. 2.3. Распределение модельных данных

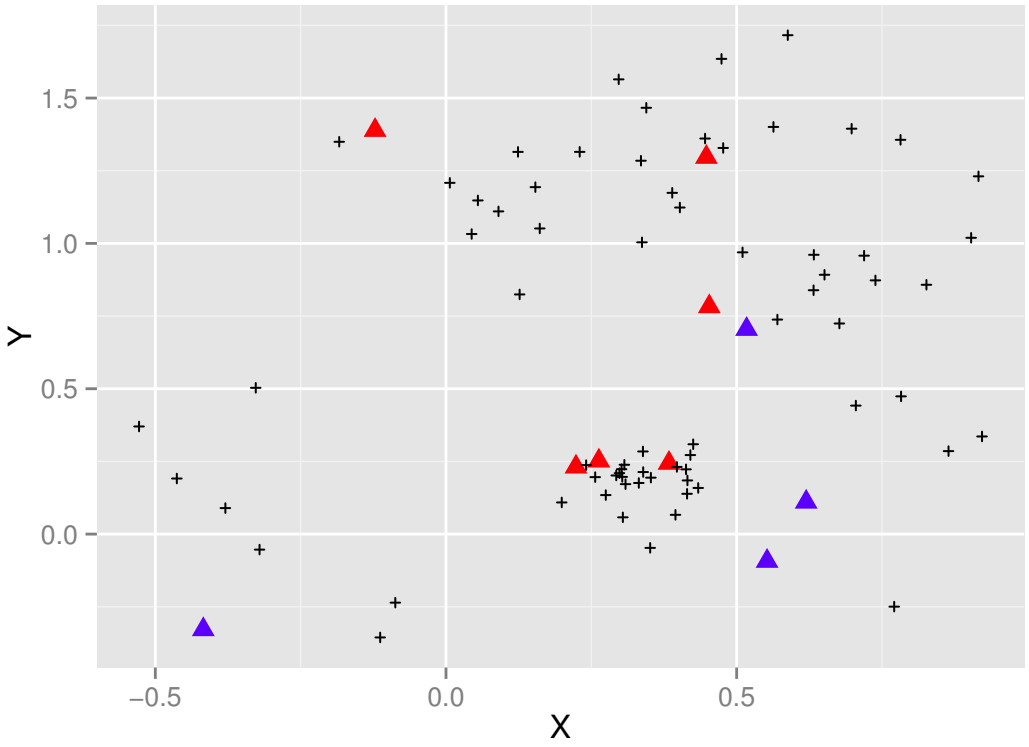

Рис. 2.4. Инициализации в модельном примере

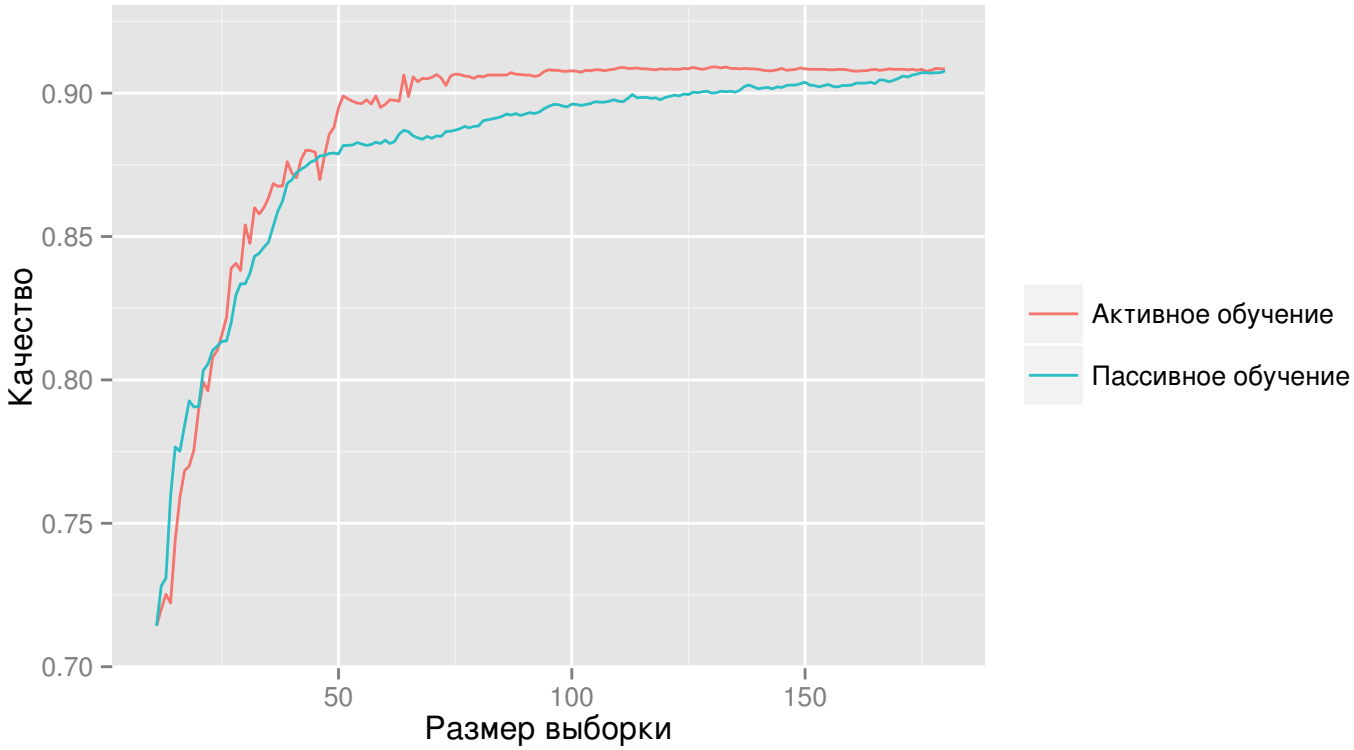

Рис. 2.5. Результат эксперимента на модельных данных

2.3.2. База MNIST

База MNIST[1] содержит более 50000 образцов рукописных цифр в градациях серого, размером 28x28 пикселей. В базе представлены объекты 10 классов в 784-мерном признаковом пространстве. Все эксперименты проводились на 20-мерных данных, понижение размерности было осуществленно с помощью метода главных компонент.

Для бинарного случая объекты класса «3» получали метку «+», все остальные - «-». Параметры эксперимента: $|\mathcal{U}| = 8000$, $|\mathcal{L}| = 100$, $C = 8000$, за одну итерацию добавлялись 100 точек. Результаты приведены на рис.2.6.

В многоклассовой постановке было проведено 2 эксперимента - с существенным ограничением на бюджет ($|\mathcal{U}| \gg C$) и с полным использованием пула доступных точек ($|\mathcal{U}| = C$).

В первом эксперименте взято большое множество $\mathcal{U}$ ($|\mathcal{U}| = 40000$), началь-

[1] доступна по адресу http://yann.lecun.com/exdb/mnist/index.html

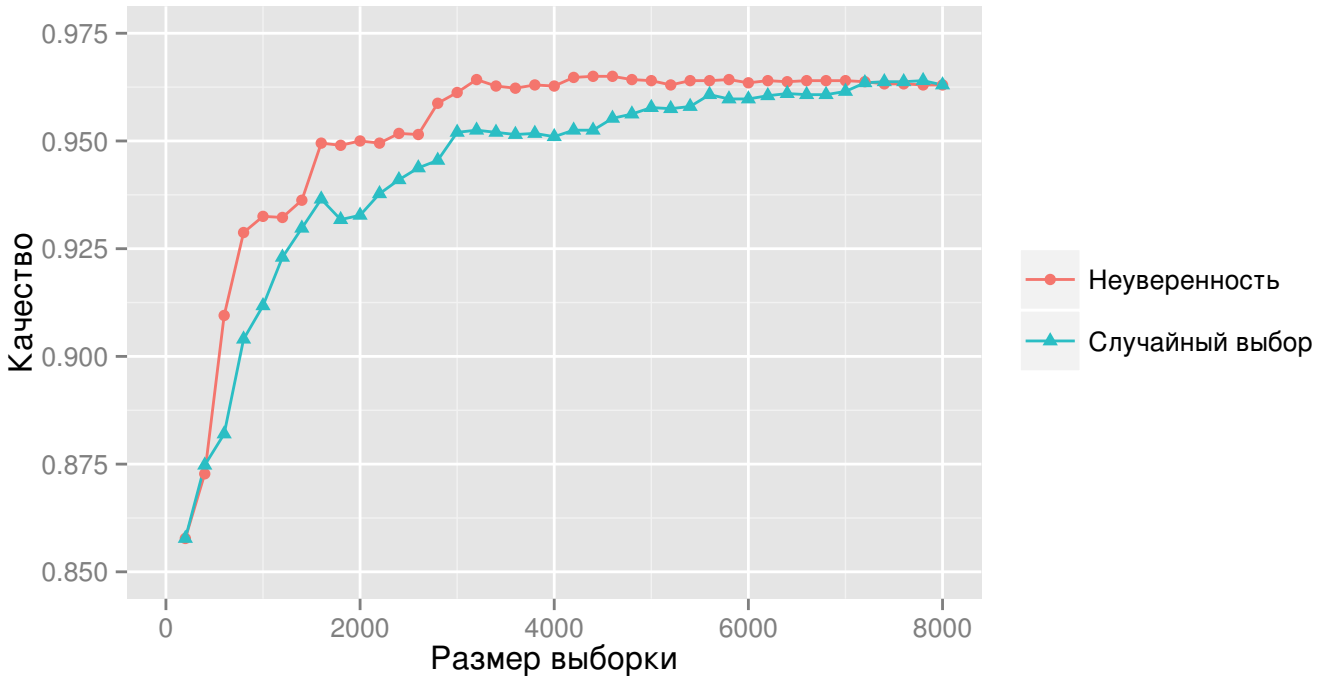

Рис. 2.6. Результат эксперимента на базе MNIST, бинарная классификация

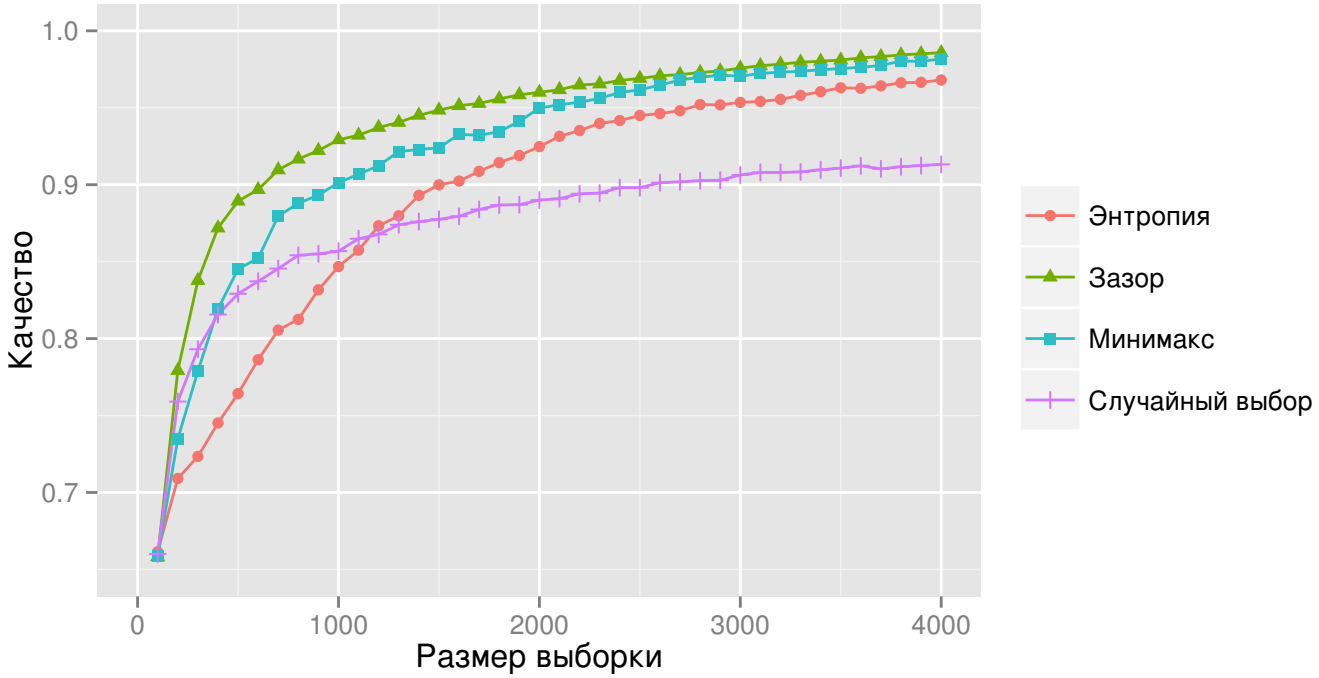

Рис. 2.7. Результат эксперимента на базе MNIST, случай $|\mathcal{U}| \gg C$

ное множество $\mathcal{L}$ размера 100 и ограничивающая константа $C = 4000$. За одну итерацию в обучающую выборку добавлялись 100 точек, результат такого эксперимента приведен на рис. 2.7

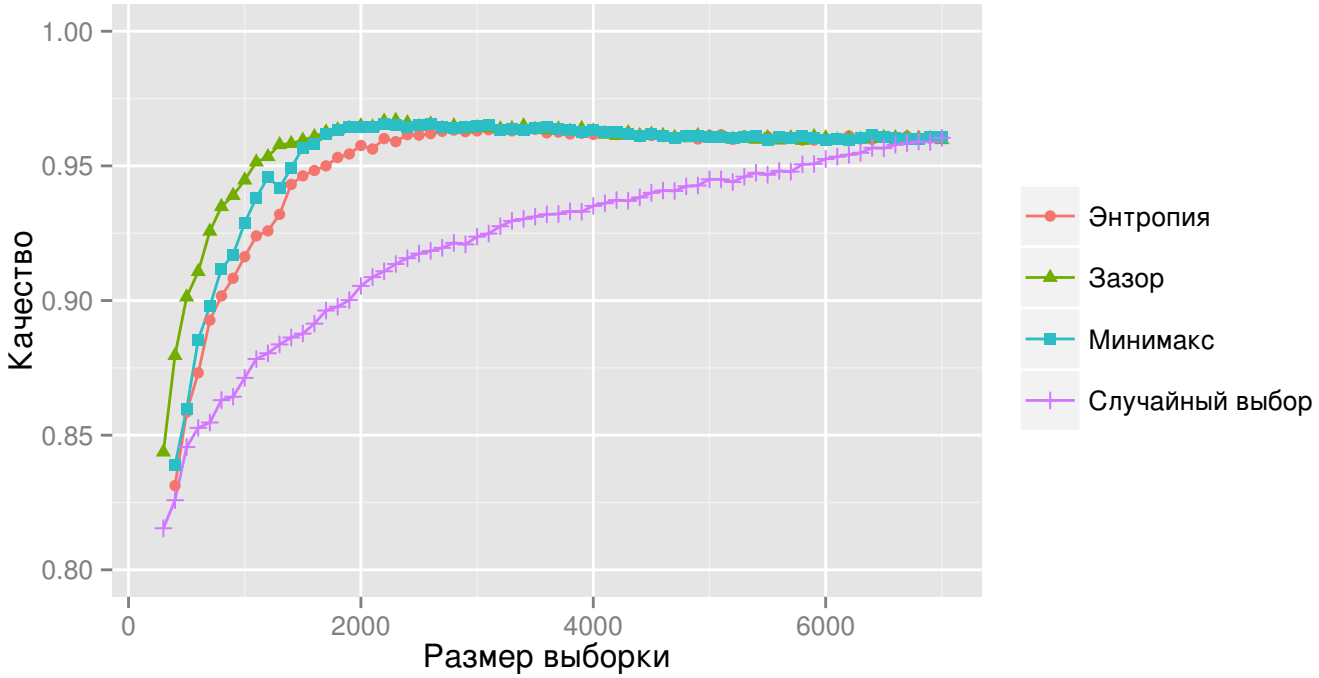

Рис. 2.8. Результат эксперимента на базе MNIST, случай $|\mathcal{U}| = C$

Во втором эксперименте $|\mathcal{U}| = 7000$, $|\mathcal{L}| = 100$, $C = 7000$. За одну итерацию также добавлялись 100 точек, результат эксперимента на рис. 2.8

2.3.3. База UCI:CoverType

База [2] представляет собой описание лесных массивов на севере Колорадо. Представлены описания 7 типов лесов, используются 54 признака. В эксперименте было взято множество $\mathcal{U}$ ($|\mathcal{U}| = 5000$), начальное множество $\mathcal{L}$ размера 100 и ограничивающая константа $C = 5000$. За одну итерацию в обучающую выборку добавлялись 200 точек, результат такого эксперимента приведен на рис. 2.9.

2.3.4. Данные компании Smart Engines

База, предоставленная компанией Smart Engines, представляет собой 500000 изображений заглавных печатных букв английского алфавита размером 16×16 пикселей. Пример данных представлен на рис. 2.11.

[2] доступна по адресу https://archive.ics.uci.edu/ml/datasets/Covertype

Параметры эксперимента: $|\mathcal{U}| = 500000$, $|\mathcal{L}| = 20000$, $C = 500000$. За одну итерацию добавлялись 20000 точек. Результат приведен на рис. 2.10.

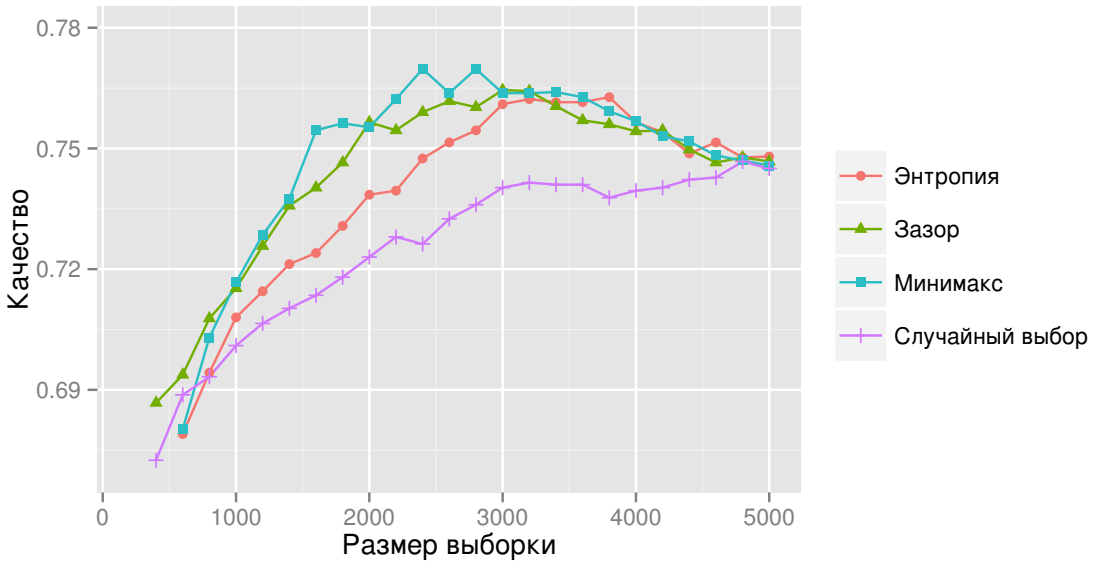

Рис. 2.9. Результат эксперимента на базе UCI:CoverType

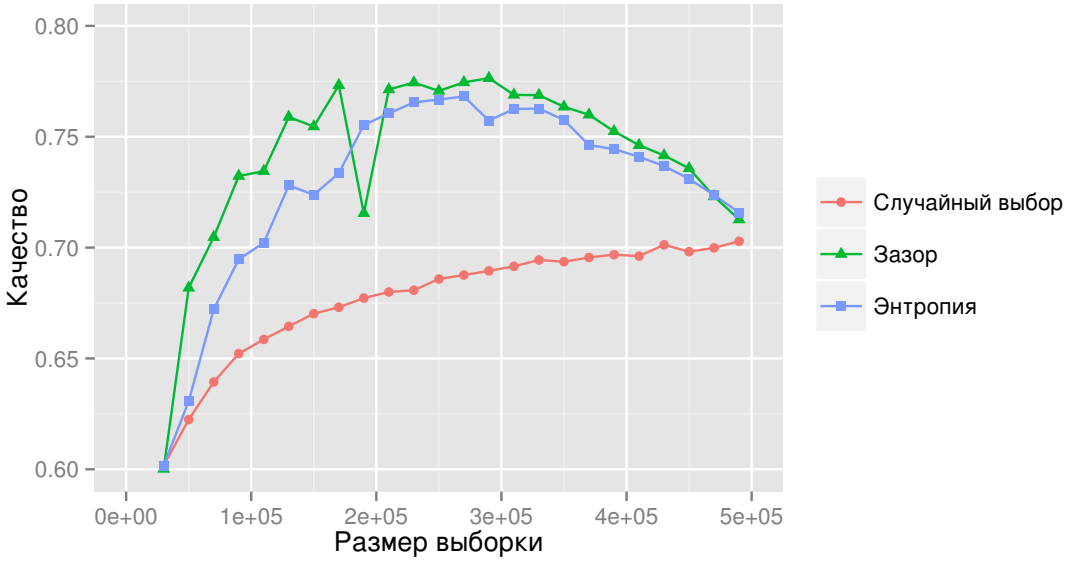

Рис. 2.10. Результат эксперимента на данных компании Smart Engines

Рис. 2.11. Пример данных из базы компании Smart Engines

2.4. Эффект отбора подмножества

Обратим внимание на случай, когда $|\mathcal{U}| = C$. Ему соответствуют графики 2.8, 2.9 и 2.10, из которых можно заметить, что пик качества достигается не при обучении на всем множестве $\mathcal{U}$, а лишь на некотором его подмножестве!

Обнаруженный эффект показывает осмысленность следующей постановки. Зафиксируем семейство базовых классификаторов $\mathcal{H}$. Пусть для $\forall x \in \mathcal{U}$ известно значение $label(x)$ или, что эквивалентно, $C = |\mathcal{U}|$. Будем искать $\mathcal{L}^* \subset \mathcal{U}$ такое, что

$$\mathcal{L}^* = \arg\max_{\mathcal{L} \subset \mathcal{U}} acc(h_{\mathcal{L}})$$

Для ее решения предлагается использовать алгоритм активного обучения и стратегию margin sampling, как показавшую наилучшие результаты.

2.4.1. Численные эксперименты

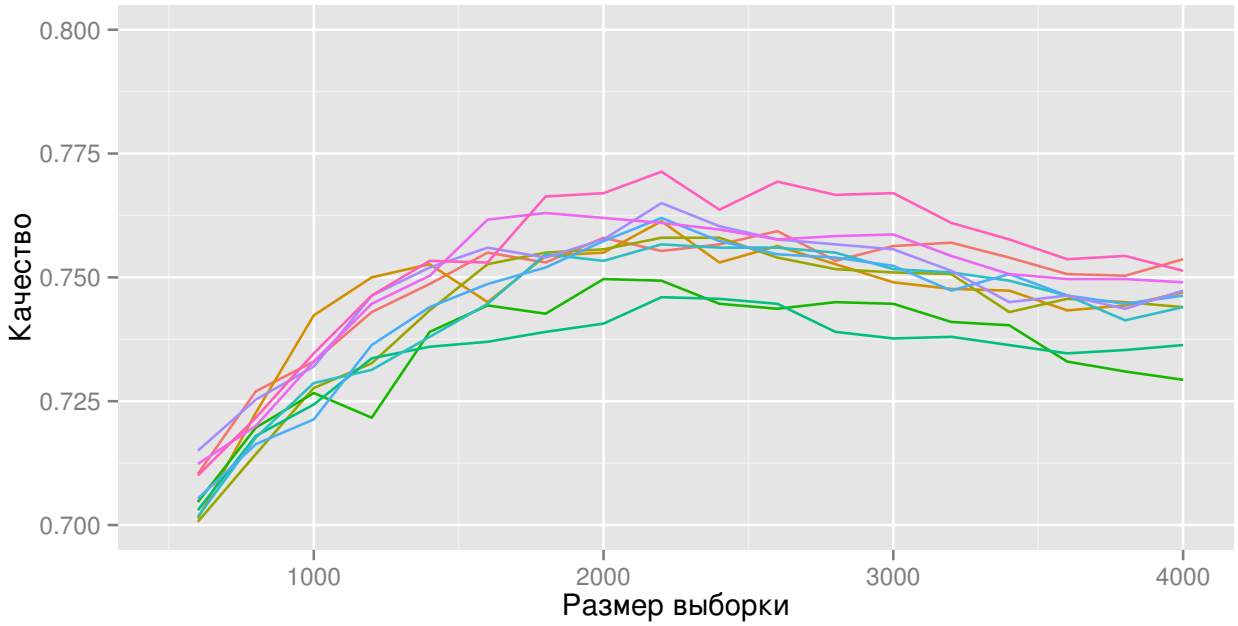

Рис. 2.12. 10 реализаций процедуры итеративного наращивания обучающего множества при использовании базы UCI:CoverType

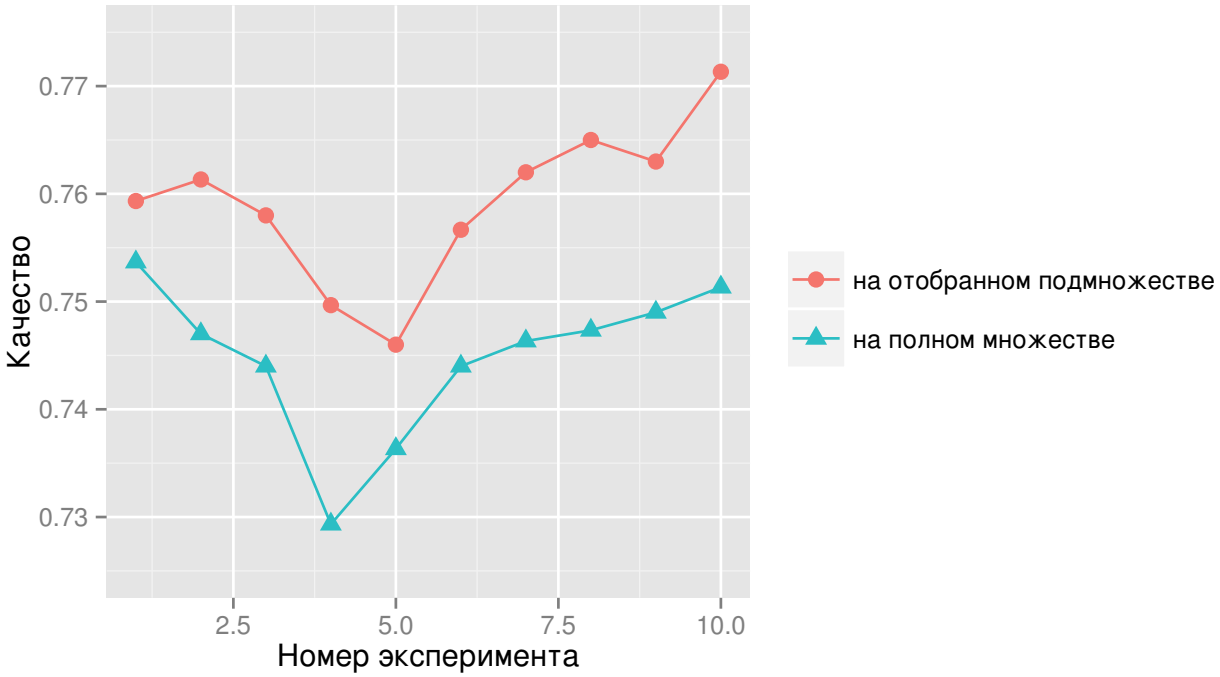

Рис. 2.13. Оценки качества при обучении на построенном подмножестве $\mathcal{L}^*$ и на множестве $\mathcal{U}$. База UCI:CoverType

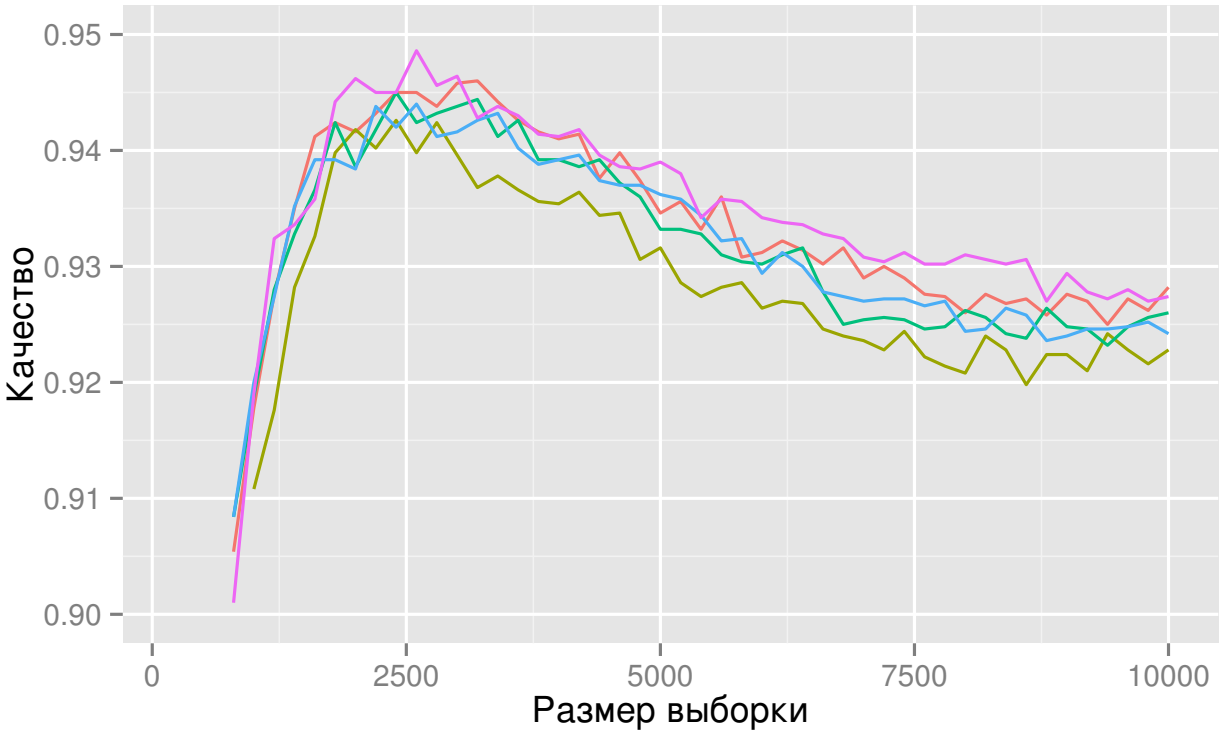

Рис. 2.14. 5 реализаций процедуры итеративного наращивания обучающего множества при использовании базы MNIST

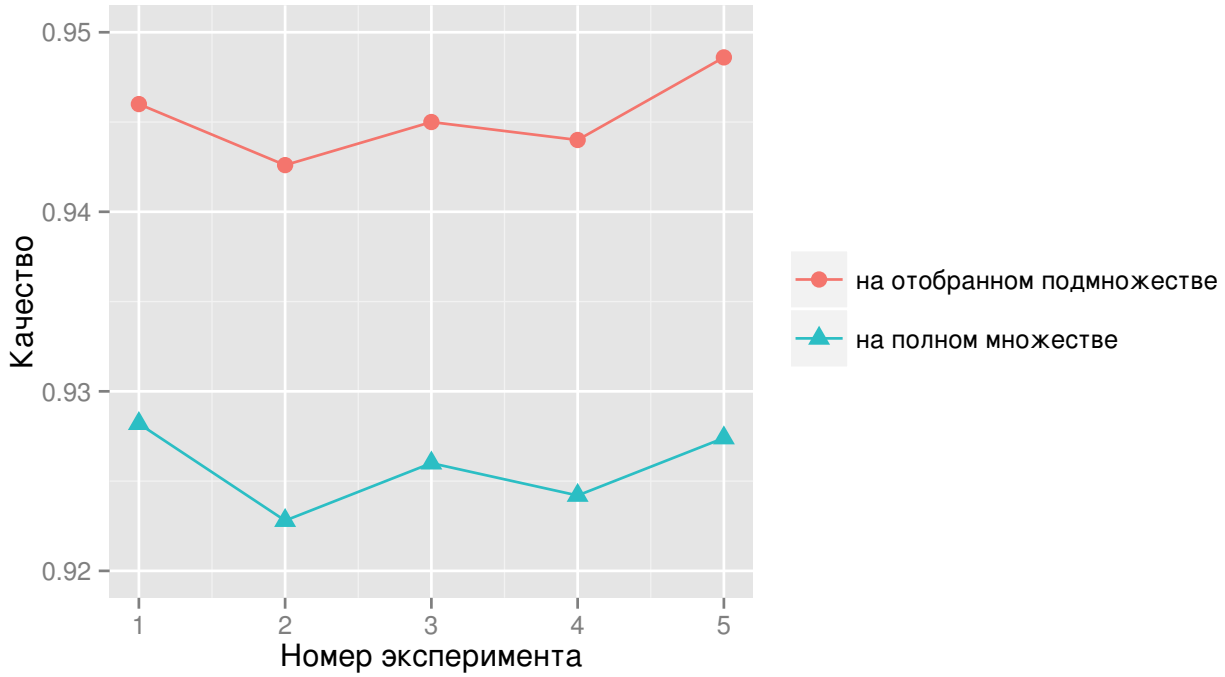

Рис. 2.15. Оценки качества при обучении на построенном подмножестве $\mathcal{L}^*$ и на множестве $\mathcal{U}$. База UCI:CoverType

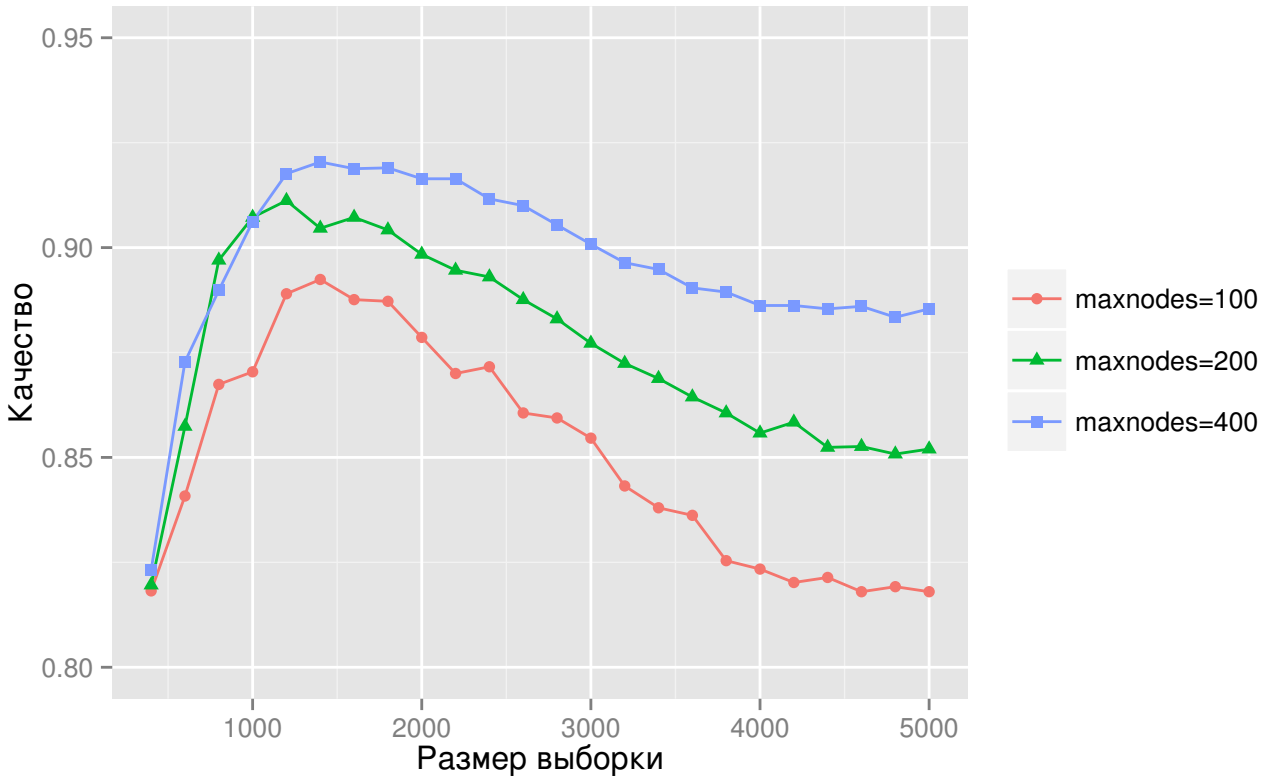

Рис. 2.16. Эксперимент с моделями фиксированной сложности, база MNIST

На каждой итерации эксперимента сэмплировалось множество $\mathcal{U}$ фиксированного размера и применялась процедура итеративного наращивания множества методом активного обучения с использованием стратегии зазора. Несколько визуализаций показаны на рис. 2.12 и рис. 2.14. Результаты приведены на рис. 2.13 и рис. 2.15. График 2.10 так же может быть проинтерпретирован как подтверждение эффекта.

Кроме того, был проведен эксперимент с моделями фиксированной сложности. Для этого использовалось ограничение на количество терминальных нод при построении каждого решающего дерева в модели случайного леса, задаваемое параметром maxnodes. Результат представлен на рис. 2.16.

2.4.2. Предлагаемый алгоритм

Полученные выше результаты показывают, что сведение традиционной задачи классификации к задаче активного обучения позволяет ожидать увеличение качества при уменьшении обучающей выборки. Ниже приводится общий вид алгоритма для использования этого эффекта.

Исходные параметры: множество $\mathcal{U}$ и модель классификатора h

Результат: множество $\mathcal{L}^*$

1 инициализируем $\mathcal{L}^*$; $i = 0$; $C = |\mathcal{U}|$;

2 **до тех пор, пока** $|\mathcal{L}^*| \leq C$ **выполнять**

3 $h(x) \leftarrow h_{\mathcal{L}^*}(x)$;

4 оцениваем $acc(h)$;

5 сохраняем пару $(\mathcal{L}^*, acc(h))$;

6 $x = \arg\max_{x \in \mathcal{U}} \phi_h(x)$;

7 $\mathcal{L}^* \leftarrow \mathcal{L}^* \cup \{(x, label(x))\}$; $\mathcal{U} \leftarrow \mathcal{U} \setminus \{x\}$;

8 $i = i + 1$;

9 **конец цикла**

10 возвращаем $\mathcal{L}^*$, соотвествующее максимальному $acc(h)$

Алгоритм 3: Общий алгоритм использования обнаруженного эффекта

Заметим, что с вычислительной точки зрения наибольшей проблемой является множественная перенастройка базового классификатора. Один из возможных подходов к ее решению - использование критерия раннего останова, опирающегося на детекцию падения качества. Кроме того, остается открытым вопрос того, как оценивать $acc(h)$. В данной работе для этого использовалось заранее выделенное тестовое множество, другой возможный способ заключается в использовании скользящего контроля. Детальное рассмотрение тонкостей реализации и оптимизации алгоритма остается за рамками данной работы.

Кроме того, отметим важность выбора базового классификатора. Была предпринята попытка обучить на отобранном в результате эксперимента на данных компании Smart Engine подмножестве нейронную сеть. В эксперименте использовалась модель «случайный лес» и было отобрано подмножество мощностью порядка 250000. Посимвольное качество нейронной сети, обученной на нем, оказалось на 1% ниже качества сети, обученной на всему множеству. Это подтверждает, что подмножество отбирается в соотвествии с базовым классификатором.

Глава 3

Стратегии поиска в пространстве гипотез

Пусть $\mathcal{H}$ - пространство классификаторов. Рассмотрим случай разделимых классов, то есть предполагаем, что $\exists\, h \in \mathcal{H} : h^*(x) = label(x) \quad \forall x \in \mathcal{U}$. Определим $\mathcal{V} \subset \mathcal{H}$ - множество гипотез, согласующихся с обучающей выборкой, то есть $\mathcal{V} = \{h \in \mathcal{H} : h(x) = y \quad \forall x \in \mathcal{L}\}$. Тогда искомый классификатор принадлежит этому множеству, то есть $h^* \in \mathcal{V}$. Идея рассматриваемого подхода состоит в выборе таких $x \in \mathcal{U}$, которые как можно сильнее будут сужать множество $\mathcal{V}$.

3.1. Описание подхода

Для дальнейших рассуждений определим понятие региона рассогласованности $DIS(\mathcal{V})$ (англ: region of disagreement) следующим образом:

$$DIS(\mathcal{V}) = \{x \mid \exists\, h_1, h_2 \in \mathcal{V} : h_1(x) \neq h_2(x)\}$$

Существует алгоритм CAL([15], [4]), который формулируется для потокового сценария и в общем виде представлен в листинге 4. В работах [16], [6] для этого алгоритма получены асимптотические оценки на число требуемых точек для достижения заданного уровня качества. Однако с практической точки зрения алгоритм CAL обладает существенными недостатками: пространство гипотез $\mathcal{V}$ может быть непрерывным (как следствие, может быть трудоемко вычислять $DIS(\mathcal{V})$) и используется только бинарная мера рассогласованности.

На практике используется подход, описанный в работе [5]. $\mathcal{V}$ аппроксимируется несколькими гипотезами, и уже по ним вычисляется оценка $\widehat{DIS}(\mathcal{V})$. Такая аппроксимация позволяет пользоваться алгоритмом в случае неразделимых выборок, но при этом лишает теоретических гарантий. Кроме того, вводит-

Исходные параметры: множество гипотез $\mathcal{V}$, $\{x_i\}_{i=1}^{\infty}$

Результат: классификатор h

$m \leftarrow 0, t \leftarrow 0, \mathcal{V} \leftarrow \mathcal{H}$

до тех пор, пока $t < n, m < 2^n$ **выполнять**

  если $x_m \in DIS(\mathcal{V})$ **тогда**

    запросить метку y_m

    $\mathcal{V} \leftarrow \{h \in \mathcal{V} : h(x_m) = y_m\}$

    $t \leftarrow t + 1$

  конец условия

конец цикла

вернуть $\forall h \in \mathcal{V}$

Алгоритм 4: алгоритм CAL для потокового сценария

ся функция рассогласованности $d(x)$, обладающая следующими свойствами:

$$d(x, h_1, \ldots, h_q) : \mathbb{R}^m \times \mathcal{H}^q \to \mathbb{R}$$

$$d(x_a, h_1, \cdots, h_q) \geq d(x_b, h_1, \cdots, h_q) \quad \forall x_a \in DIS(\mathcal{V}) \quad \forall x_b \in \overline{DIS(\mathcal{V})}$$

Введем несколько различных функций рассогласованности. Пусть есть q гипотез, когда $d(x)$ будет оценивать число уникальных предсказаний. Формальное определение:

$$d^{unique}(x, h_1, \cdots, h_q) = |\{h_1(x), h_2(x), \cdots, h_q(x)\}|$$

Другую меру рассогласованности можно получить, если оценивать ее как количество попарно несовпавших предсказаний. Формально:

$$d^{pairwise}(x, h_1, \cdots, h_q) = |\{(i, j) \in \{1, \ldots, q\}^2 : h_i(x) \neq h_j(x)\}|$$

Можно использовать не предсказания, а получаемые оценки вероятностных распределений принадлежности точки к классам. В этом случае для оценки рассогласованности предлагается подсчитать среднее всех попарных расстояний

Кульбака — Лейблера:

$$d^{KL}(x, h_1, \cdots, h_q) = \frac{1}{|\mathcal{U}|} \sum_{i,j}^{q} KL(prob_{h_i}(x), prob_{h_j}(x))$$

где $KL(P, Q)$ - расстояние Кульбака — Лейблера между дискретными распределениями P и Q.

3.2. Численный эксперимент

В качестве аппроксимации $\mathcal{V}$ использовались q классификаторов, обученных на q случайных подмножествах $\mathcal{L}_i^* \subset \mathcal{L}$, причем $|\mathcal{L}_i^*| = 0.75|\mathcal{L}|$ $\forall i = 1, \ldots, q$. В качестве мер неопределенности рассмотрены приведенные выше функции d^{unique}, $d^{pairwise}$ и d^{KL}. Заметим, что рассматриваемый подход может быть помещен в рамки общего алгоритма активного обучения (листинг 1), если рассматривать обучение модели как настройку ансамбля из q классификаторов, а функцию критерия определить как $\phi_{\{h\}}(x) = d(x, \{h\})$

3.2.1. База UCI:CoverType

Описание этой базы дано в секции 2.3.3. Отметим только, что в эксперименте использовались следующие ограничения: $|\mathcal{U}| = 5000$, $|\mathcal{L}| = 200$ и ограничивающая константа $C = 5000$. Число моделей в ансамбле (q) равно 10. За одну итерацию в обучающую выборку добавлялись 200 точек. Результат такого эксперимента приведен на рис. 3.1

3.2.2. База MNIST

Описание этой базы дано в секции 2.3.2. Было проведено 3 эксперимента - с разным шагом алгоритма, размером пула и размером ансамбля.

В первом случае было взято $|\mathcal{U}| = 4000$, $|\mathcal{L}| = 200$, $C = 3000$. За одну итерацию в обучающую выборку добавлялись 200 точек, результат такого эксперимента приведен на рис. 3.2

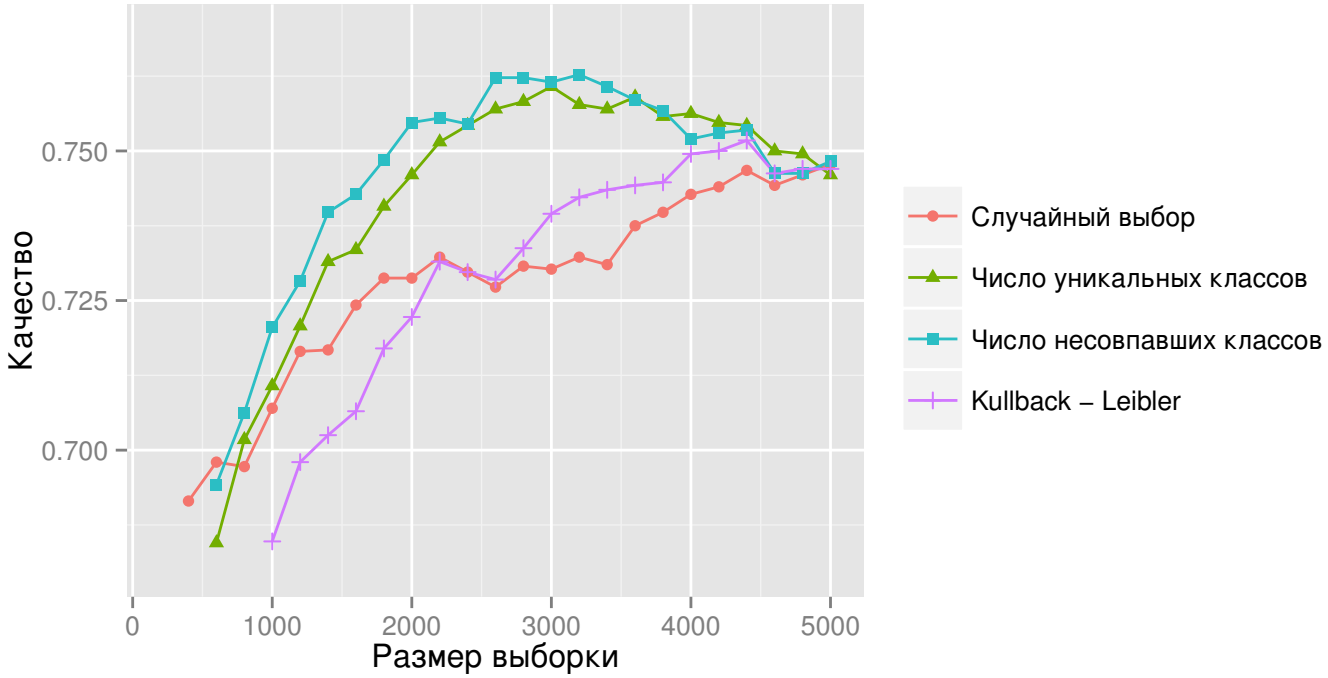

Рис. 3.1. Результат эксперимента на базе UCI:CoverType

Во втором случае был взята более мелкий шаг алгоритма и больший пул. $|\mathcal{U}| = 7000$, $|\mathcal{L}| = 100$, $C = 7000$. За одну итерацию также добавлялись 100 точек, результат эксперимента на рис. 3.3.

В третьем эксперименте оценивалось качество ансамблей разных размеров. В качестве функции рассогласованности использовалась $d^{pairwise}$. Результаты приведены на рис. 3.4.

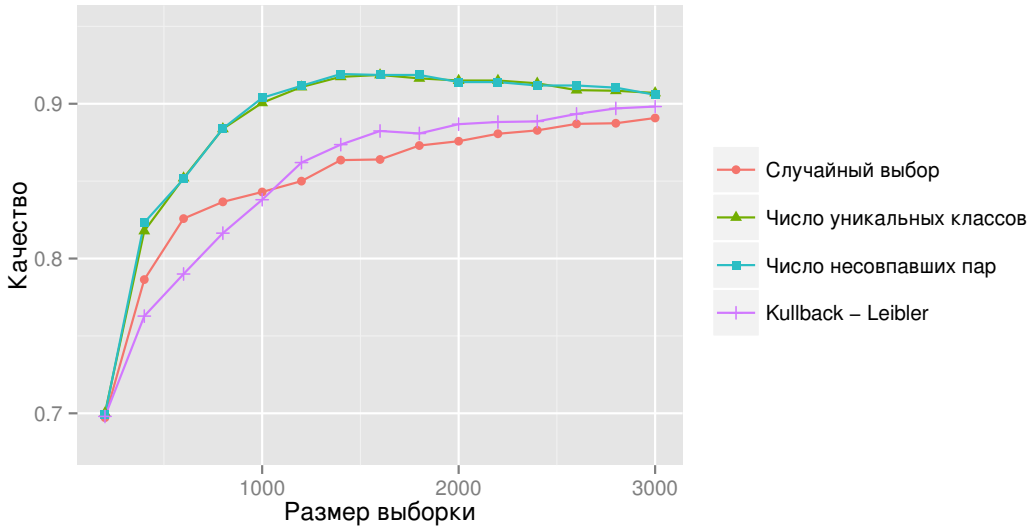

Рис. 3.2. Результат эксперимента на базе MNIST, шаг алгоритма - 200 точек

Отметим, что лучший результат показала функция рассогласованности, которая опирается только на предсказание базовой модели, не используя значения $prob_{h_i}(x)$. Это значит, что в такой модификации алгоритм может использоваться с базовыми классификаторами типа «черный ящик», не накладывая никаких ограничений на модель.

Кроме того, на рис.3.1 и рис.3.3 видно, что пик качества достигается при обучении на подмножестве пула, а не на всем пуле, что еще раз подтверждает наличие эффекта, описанного в предыдущей главе.

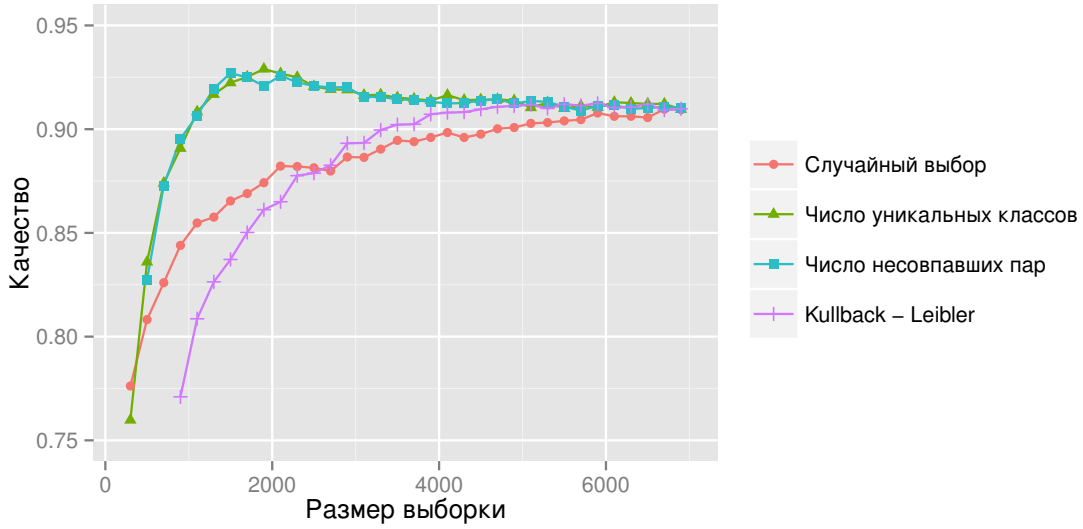

Рис. 3.3. Результат эксперимента на базе MNIST, шаг алгоритма - 100 точек

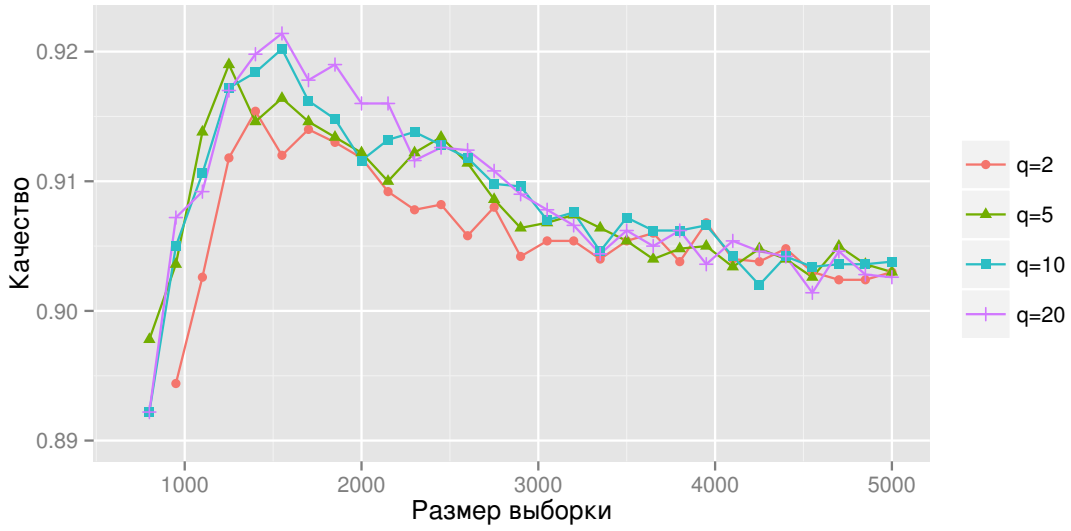

Рис. 3.4. Результат эксперимента с ансамблями разного размера на базе MNIST

Выводы

В работе был дан обзор стратегий активного обучения и проведено сравнение двух подходов. Полученные результаты показывают, что методы, основанные на эвристиках, показывают хорошие практические результаты. Стратегии, основанные на неуверенности, и стратегии, основанные на поиске в пространстве гипотез, дают сравнимые результаты. Первые - вычислительно проще, но требуют, чтобы базовая модель могла оценивать распределение вероятностей принадлежности к классам. Второй подход поддерживается некоторыми теоретическими результами в виде оценок на число требуемых размеченных примеров и не требует определения функции $prob_h(x)$, что ослабляет ограничения на модель. Платой за это является бóльшая вычислительная сложность.

Результаты показывают, что имеет смысл редукция традиционной задачи классификации к задаче активного обучения. Обнаружен эффект выделения подмножества, обучение на котором дает качество не хуже, чем обучение на всей доступной выборке. Кроме того, происходит уменьшение размера обучающей выборки, что позволяет сократить время настройки итогового классификатора. Был предложен алгоритм для использования этого эффекта, проведен эксперимент на реальных данных компании Smart Engines.

За рамками данной работы остался вопрос о том, как лучше в пакетном режиме на каждом шаге выбирать множество точек, добавляемых в обучающую выборку. Очевидно, что результаты можно улучшить, если выбирать множество с учетом взаимной информации. В работе [17] были предприняты попытки строить суррогатную модель для упрощения задачи оптимизации по подмножеству, однако вопрос остается открытым.

Список литературы

1. Settles B. Active Learning. Morgan and Claypool, 2012.
2. Guyon. Results of the Active Learning Challenge // JMLR: Workshop and Conference Proceedings 16. 2011. P. 19–45.
3. Hanneke S. Theory of Active Learning. 2014.
4. M. Balcan J. L., A. Beygelzimer. Agnostic Active Learning // Proceedings of the 23rd International Conference on Machine Learning. 2006.
5. H. S. Seung H. S., M. Opper. Query by committee // Proceedings of the fifth annual workshop on Computational learning theory. 1992. P. 287–294.
6. Hanneke S. Rates of convergence in active learning // The Annals of Statistics 39. 2011. P. 333–361.
7. Fujiwara A. Drug Screening of GPCR Using Active Learning // Genome Informatics 14. 2003. P. 597–598.
8. Lovell C. Autonomous Experimentation: Active Learning for Enzyme Response Characterisation // JMLR: Workshop and Conference Proceedings 16. 2011. P. 141–155.
9. Dasgupta S. Hierarchical Sampling for Active Learning.
10. Long J. Graph-Based Active Learning Based on Label Propagation // Modeling Decisions for Artificial Intelligence. 2008. P. 179–190.
11. Hospedales T. M. A Unifying Theory of Active Discovery and Learning // ECCV Part V, LNCS 7576. 2012. P. 53–466.
12. Haines X. Active Learning using Dirichlet Processes for Rare Class Discovery and Classification. 2011.
13. Breiman L. Random Forests // Machine Learning 45. 2001. P. 5–32.
14. Biau G. Analysis of a Random Forests Model // Journal of Machine Learning Research 13. 2012. P. 1063–1095.
15. D. Cohn R. L., L. Atlas. Improving generalization with active learning // Machine Learning 15. 1994. P. 201–221.

16. Dasgupta S. Two faces of active learning.

17. Damoulas T. AL2: Learning for Active Learning.