



зованного разреза является самым медленным, так как нахождение векторов собственных значений занимает $O(N^3)$.

В описанных выше методах быстроедействие и качество разбиения во многом зависит от параметров условия останова. В статье [3] предложено использование matrix-free алгоритмов для нахождения собственных векторов для метода нормализованного разреза, что должно существенно увеличить производительность. На практике данный алгоритм уместно применять с предварительной генерацией супер-пикселей.

Таким образом можно заключить, что каждый алгоритм порождает решение свойственное определенной задаче. Для отделения контрастного изображения от фона подходит метод минимального разреза. Для сегментации изображений с содержанием большого количества мелких деталей – эффективный метод. Для задач, где время не является критическим и требуется разделить изображение на равные части – метод нормализованного разреза.

Литература

- 1 Форсайт Д., Компьютерное зрение. Современный подход [Текст]/ Форсайт Д., Понс Ж.–Москва: Вильямс, 2004. – 928 с.
- 2 Boykov Y., Graph Cuts and Efficient N-D Image Segmentation / Boykov Y., Funka G. // Int J Comput Vision.–2006. –№ 70 –С. 109–131.
- 3 Jianbo S., Normalized Cuts and Image Segmentation / Jianbo S., Jitendra M. // IEEE Transactions on pattern analysis and machine intelligence–2000, –№ 20 – С. 888–905.
- 4 Felzenszwalb P., Efficient Graph-Based Image Segmentation / Felzenszwalb P. // Journal International Journal of Computer Vision archive –2004– № 59,–С. 167 – 181.

Д.А. Жуков, В.Н. Клячкин

АЛГОРИТМЫ БУСТИНГА В ЗАДАЧАХ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ

(Ульяновский государственный технический университет)

Для решения основной задачи технической диагностики – распознавания исправности состояния рассматриваемого объекта [1], могут быть использованы методы машинного обучения, с помощью которых проводится бинарная классификация: объекты подразделяются на исправные и неисправные с помощью моделей, полученным по обучающим выборкам [2-4].

Распознавание состояния технического объекта обычно проводится по результатам измерений косвенных показателей функционирования этого объекта в условиях ограниченной информации. В качестве исходных данных рассматриваются известные результаты оценки состояния системы: при заданных значениях контролируемых показателей техническая система исправна или неисправна. Таким образом, имеется множество N прецедентов $(x^{(i)}, y^{(i)})$, $i = 1 \dots N$:



объекты с заданными d показателями функционирования $x = (x_1, x_2, \dots, x_d)$ и соответствующими состояниями y , принимающими одно из двух значений $(-1, +1)$: $y = +1$ соответствует исправному состоянию объекта, $y = -1$ – неисправному. По существу, известна конечная совокупность пар «набор показателей, состояние» – исходная выборка данных.

В общем случае показатель $x_j, j = 1 \dots d$, может быть количественным, номинальным или порядковым (например, температура может быть пониженной, нормальной или повышенной). На основе этих данных $(x^{(i)}, y^{(i)})$ требуется восстановить зависимость между показателями функционирования и состоянием объекта, то есть построить алгоритм, способный для любого набора показателей объекта выдать достаточно точный ответ о его состоянии. Это частный случай одной из задач машинного обучения – классификации при обучении по прецедентам (с учителем). Полученный алгоритм должен обладать обобщающей способностью: обеспечивать достаточно точные прогнозы на новых показателях, не входящих в исходную выборку.

Качество классификации, которое определяет эффективность машинного обучения, зависит от ряда факторов: объема и качества исходной выборки, метода машинного обучения, способа разделения исходной выборки на обучающую и контрольную части, отбора значимых показателей и других [5-7].

Одним из наиболее эффективных методов машинного обучения является бустинг, представляющий последовательное построение композиции из «слабых» алгоритмов обучения, при котором каждый следующий алгоритм пытается компенсировать недостатки предыдущих алгоритмов (boosting – усиление). Основные причины широкого распространения этого метода – простота, универсальность, гибкость (возможность построения различных модификаций), а также высокая обобщающая способность [3,4,8].

Итоговое правило в бустинге строится путем взвешенного голосования ансамбля (композиции) базовых правил. При этом используется информация об ошибках предыдущих правил: веса объектов выбираются таким образом, чтобы новое правило точнее работало на тех объектах, на которых предыдущие правила чаще ошибались.

Классификатор может быть представлен в виде знака функции

$$H(x) = \text{sign} \left[\sum_{t=1}^T \alpha_t h_t(x) \right], \quad (1)$$

где $h_t(x)$ – базовые алгоритмы классификации, возвращающие один из двух результатов: -1 или $+1$; α_t – коэффициент взвешенного голосования для алгоритма $h_t(x)$.

Функционал качества композиции можно определить как число ошибок, допускаемых ею на заданной выборке:

$$Q = \sum_{i=1}^N L(y, H) = \sum_{i=1}^N \left[y^{(i)} \sum_{t=1}^T \alpha_t h_t(x^{(i)}) < 0 \right], \quad (2)$$



где $L(y, H)$ – функция потерь. Этот функционал необходимо минимизировать. При использовании алгоритма адаптивного усиления AdaBoost предполагается экспоненциальная аппроксимация функции потерь:

$$(3) \quad L(y, H) = \exp(-yH).$$

Алгоритм AdaBoost сводится к следующей последовательности действий: инициализируются веса наблюдений $w_i = 1/N$, организуется цикл $t = 1 \dots T$, обучается простой классификатор $h_t(x)$ и определяется его ошибка (суммируются веса ошибочно классифицированных наблюдений)

$$\varepsilon_t = \sum_{i: h_t(x^{(i)}) \neq y^{(i)}} w_i(t), \quad (4)$$

коэффициент взвешенного голосования определяется по формуле:

$$\alpha_t = \frac{1}{2} \ln \frac{1 - \varepsilon_t}{\varepsilon_t}. \quad (5)$$

Далее проводится перерасчет весов наблюдений: если наблюдение классифицируется правильно, вес уменьшается, если неправильно – вес увеличивается:

$$w_i(t+1) = w_i(t) \exp(-\alpha_t y^{(i)} h_t(x^{(i)})), \quad (6)$$

веса наблюдений нормируются; проводится переход к следующему t . По окончании работы алгоритма получаем итоговый классификатор (1).

При наличии шумовых данных в исходной выборке более эффективной может оказаться разновидность бустинга LogitBoost, основанная на использовании идеи логистической регрессии. Функция потерь (3) в этом случае имеет вид:

$$L(y, H) = \log(1 + \exp(-2yH)). \quad (7)$$

Минимизация функционала (2) возможна методом градиентного спуска (градиентный бустинг), это наиболее общий случай бустинга. Выбор конкретного метода определяется особенностями исходной выборки.

Для оценки качества обучения (обобщающей способности классификатора) исходная выборка разбивается на обучающую и контрольную. Иногда используют метод скользящего контроля, при котором множество N прецедентов разбиваются на q частей (на практике часто принимают $q = 10$). В качестве обучающей выборки используют $(q - 1)$ частей, а оставшуюся часть выборки – для контроля. Средняя ошибка по контрольным выборкам на всех возможных разбиениях характеризует обобщающую способность полученного классификатора.

Литература

1. Биргер, И. А. Техническая диагностика / И. А. Биргер. – М.: Машиностроение, 1978. – 240 с.



2. Жуков, Д.А. Задачи обеспечения эффективности машинного обучения при диагностике технических объектов / Д.А.Жуков, В.Н.Клячкин // Современные проблемы проектирования, производства и эксплуатации радиотехнических систем. – 2016. – № 1 (10). – С. 172-174.

3. Witten, I. H. Data mining : practical machine learning tools and techniques / Ian H.Witten, Eibe Frank. – 2nd ed. – San Francisco: Morgan Kaufmann Publishers, 2005. – 525p.

4. Мерков, А. Б. Распознавание образов. Введение в методы статистического обучения / А.Б. Мерков. – М.: Едиториал УРСС, 2011. – 256 с.

5. Клячкин, В.Н. Статистические методы анализа данных / В.Н. Клячкин, Ю.Е. Кувайскова, В.А. Алексеева. – М. : Финансы и статистика, 2016. – 240 с.

6. Клячкин, В.Н. Применение методов машинного обучения при решении задач технической диагностики / В.Н.Клячкин, И.Н. Карпунина, Ю.Е.Кувайскова, А.С.Хорева // Научный вестник УИГА. – 2016. – Т. 8. – С. 158-161.

7. Шунина, Ю.С. Прогнозирование платежеспособности клиентов банка на основе методов машинного обучения и марковских цепей / Ю.С. Шунина, В.Н. Клячкин // Программные продукты и системы. – 2016. – № 2. – С. 105-112.

8. Воронцов К.В. Машинное обучение. Композиции классификаторов. <http://www.intuit.ru/studies/courses/13844/1241/lecture/27000>

И.В. Ковалева, Р.И. Баженов

РАЗРАБОТКА ДВУХМЕРНОЙ ИГРЫ В СИСТЕМЕ ТРЕХМЕРНОГО МОДЕЛИРОВАНИЯ UNITY3D

(Приамурский государственный университет имени Шолом-Алейхема)

В настоящее время, компьютерные игры все больше становятся популярными. Но есть игры, в которые можно играть только платно. Поэтому существует возможность создать свою компьютерную игру и для этого использовать платформу разработки для создания многоплатформенных 2D и 3D игр – Unity3D. Процесс разработки довольно прост, а результат — глубоко оптимизирован, и его можно разворачивать на других платформах, которых огромное количество.

Система Unity3D становится, все больше, популярна среди ученых, а так же и среди студентов. О.С. Ходос и Р.И. Баженов применили трехмерное моделирование в Unity3D [1]. Ф.Р. Аметов, И.Ш. Мевлют разработали пользовательский интерфейс игры на платформе unity [2]. А.С. Винокуров и Р.И. Баженов создали проект «Танк на острове» в Unity3D[3], а так же А.А. Ковтун и Е.Ю. Тихонов разработали игровое приложение «Paperman» [4]. Есть и те, кто занимался созданием не только компьютерных игр, но и созданием мобильных игр и приложений. Например, А.С. Сеидова и В.С. Сухоплюева разработали мо-