Компьютерные инструменты в образовании, 2016
№ 3: 16–24
УДК: 004.9
http://ipo.spb.ru/journal

МЕТОДЫ БАЛАНСИРОВКИ И НОРМАЛИЗАЦИИ ДАННЫХ ДЛЯ УЛУЧШЕНИЯ КАЧЕСТВА КЛАССИФИКАЦИИ

Никулин Владимир Николаевич, Канищев Илья Сергеевич,
Багаев Иван Владимирович

Аннотация

Очень часто непосредственное использование стандартных моделей приводит к результатам низкого качества. В статье рассмотрены два примера. Первый пример касается классификации популярных данных “Credit”, полученных с платформы Kaggle. В качестве классификатора мы используем стандартную функцию $nnet$ (нейронные сети) в программной среде R. Проблема состоит в том, что данные “Credit” являются несбалансированными, а функция $nnet$ склонна игнорировать класс, который составляет меньшинство. В качестве решения проблемы несбалансированности мы предлагаем рассмотреть большое число относительно небольших и сбалансированных подмножеств, в которых элементы из тренировочной базы данных отбираются случайным образом. Второй пример касается широкоизвестных данных MNIST при использовании стандартной функции svm (метод опорных векторов) в среде Python. Показана необходимость нормализации исходных признаков.

Ключевые слова: *машинное обучение, анализ данных нейронные сети, однородное ансамблирование, несбалансированность данных, распознавание образов, метод опорных векторов.*

ВВЕДЕНИЕ

Анализ данных (data mining), или вычислительная статистика, является сравнительно новой научной областью. Важность анализа данных (АД) обусловлена огромными базами данных, которые порождаются передовыми технологиями. В настоящее время вычислительная статистика является одной из наиболее динамичных и перспективных научных направлений. Успешные результаты в области АД базируются на глубоком понимании статистических закономерностей и на свободном владении языками научного программирования. Отметим, что в отличие от теоретической статистики, вычислительная статистика ориентирована на обработку и анализ реальных баз данных. Соответственно, центральным вопросом является оценивающий критерий, согласно которому производится сравнение различных методов и алгоритмов.

Идея проведения соревнований по анализу данных была впервые предложена и реализована организаторами международной конференции KDD (Knowledge Discovery in Databases) в 1998 году. По ряду причин эта идея является очень перспективной [2]. Представляется нереалистичным ожидать, что одна группа учёных могла бы включать всё разнообразие и всю многоплановость знаний и опыта множества команд из различных стран. С другой стороны, оценка качества результатов является независимой.

Формирование достаточно больших баз данных для экспериментов, основанных на реальных наблюдениях, является чрезвычайно важной и трудоёмкой задачей. Организаторы соревнований располагают всеми необходимыми специфическими качествами для успешного выполнения этой задачи. Область приложения здесь не ограничена и включает экономику, финансы, медицину, экологию, спорт и образование. Как это хорошо известно, практический опыт является лучшим способом обучения, и участие в соревнованиях по вычислительной статистике может быть очень полезно для научных работников, прикладников и, в особенности, для студентов.

В настоящее время все более популярными становятся дистанционные соревнования и конкурсы. Участвовать в таких конкурсах можно в комфортной обстановке у себя дома или в школе, а организаторам требуется один раз автоматизировать процесс, после чего их основной обязанностью остается только подготовка заданий и, в случае творческих конкурсов, оценка работ участников [1].

В настоящее время множество данных доступно посредством интернета. В частности, в наших курсах по АД мы используем данные Credit (финансовый риск кредитования) и MNIST (распознавание рукописных цифр) [5]. Мы делим наблюдения случайным образом на три части (триплет): 1) тренировка, 2) валидация (самоконтроль) и 3) тестирование, где третий случай используется для проверки знаний и навыков, поэтому метки не предоставляются. Студенты (в составе небольших групп) анализируют отдельный триплет. Оценка всех произведённых решений осуществляется в автоматическом режиме. Процесс обучения проходит в форме локального соревнования, что особенно стимулирует студентов изучать наиболее передовые методы машинного обучения. В наших курсах мы используем следующие платформы и языки программирования: R, Matlab, Python и JAVA/Weka.

Эксперименты

Отметим, что для успешного интеллектуального анализа очень важна правильная интерпретация данных. В частности, необходомо корректное представление категориальных данных. А также, имеет смысл рассмотреть гистограммы, соответствующие различным признакам с тем, чтобы выявить значительные отклонения от нормальных значений. Отдельные наблюдения, имеющие существенные отклонения, могут оказать значительное влияние на качество работы классификатора. В некоторых случаях, полное удаление некоторых признаков приводит к улучшению модели.

В последующих разделах мы рассмотрим некоторые интересные закономерности, выявленные в ходе наших экспериментов. В частности, нами было замечено, что прямое использование стандартных программных инструментов в отношении классических баз данных может привести к результатам низкого качества.

1. ДАННЫЕ CREDIT

Популярные данные Credit касаются банковского кредитования и являются несбалансированными: проблемные клиенты составляют 6,7%. Следует отметить, что стандарная функция *nnet* имеет склонность игнорировать класс-меньшинство, который и является основной целью моделирования. Однако мы можем перенести рассмотрение на последовательность случайно отобранных сбалансированных подмножеств, где фи-

нальная решающая функция вычисляется как арифметическое среднее отдельных локальных решений. Как следствие, качество классификации улучшается многократно.

1.1. Проблема несбалансированности данных при обучении нейронных сетей

Рассмотрим матрицу X признаков или объясняющих переменных и вектор меток $Y = \{1, 2, ..., k\}$, каждый i-й элемент которого соответствует i-й строке матрицы X. В этом случае мы имеет задачу классификации с k классами. По определению, данные являются несбалансированными, если пропорции элементов X с различными значениями Y существенно отличаются.

В качестве примера рассмотрим данные “Give Me Some Credit”. Впервые эти данные были опубликованы в рамках одноименного соревновения, проводимого на платформе Kaggle осенью 2011 года. Цель состояла в построении на основе исторических данных наиболее эффективного классификатора, который позволил бы определить кредитоспособность клиента в ближайшие девяносто дней. По тем временам конкурс оказался одним из наиболее популярных и привлёк около тысячи команд-участниц со всего мира.

Анонимизированные данные соответствуют двумстам пятидесяти тысячам клиентов, из которых сто тысяч с неизвестными метками были отведены для тестирования и подведения итогов соревнования. Объясняющие переменные включают в себя информацию о возрасте, семье, доходе и кредитной истории заёмщика. Целевая переменная, или метка, в данном случае принимает два значения (бинарный случай): 0 — для «хороших» клиентов, не имеющих проблем с выплатой кредита, 1 — для «плохих», имеющих определённый срок просроченной задолженности. Причём пропорции этих категорий в базе данных существенно отличаются — 93,3% и 6,7% соответственно, что говорит о сильной несбалансированности тренировочных данных.

В качестве критерия оценки эффективности алгоритмов использовался критерий AUC (area under receiver operating curve), где показатель AUC — площадь под характеристической кривой (ROC). По определению, ROC-кривая показывает пропорцию верно классифицированных положительных наблюдений (True Positive Rates) как функцию пропорции неверно классифицированных положительных примеров (False Positive Rates).

Отметим, что несбалансированность данных негативно сказывается на работе нейронных сетей — алгоритм игнорирует малочисленный класс, что приводит плохому результату классфикации. В частности, при построении модели на всей обучающей выборке (например при помощи стандартной функции $nnet$[1] в среде R) мы получим тривиальное нулевое решение с соответствующим результатом $AUC = 0,5$, что говорит о крайне низком качестве разделения классов. Решением этой проблемы является перенос рассмотрения со всей базы данных на большое количество случайно отобранных и сбалансированных подмножеств [3]. Существенной особенностью метода является тот факт, что оставщиеся данные (после вычета сбалансированного подмножества для локального обучения модели) могут быть использованы для локального тестирования модели. Таким образом, модель со случайно отобранными сбалансированными подмножествами естественным образом сочетает два процесса: 1) обучения и 2) тестирования. Так, при обучении нейросети на одном из подмножеств мы получили довольно высокий результат классификации $AUC = 0,84$.

[1] https://cran.r-project.org/web/packages/nnet/index.html

1.2. Метод случайных сбалансированных подмножеств

Для решения проблемы несбалансированности исходных данных мы предлагаем использовать метод со случайно сбалансированными подмножествами. Суть данного метода состоит в построении большого количества локальных классификаторов на основе случайно сформированных сбалансированных подмножеств. Следует отметить, что при вычислении каждого из локальных классификаторов мы искусственно снижаем количество наблюдений из преобладающего класса с целью сделать обучающую выборку более сбалансированной. В силу своей стохастичности, отдельно взятое подмножество не позволяет объективно оценивать качество обучаемого на нём классификатора, поэтому представляется целесообразным использовать технику однородного ансамблирования и вычислять решающую функцию как арифметическое среднее большого числа решений, полученных на каждом отдельно взятом подмножестве [6].

Такой подход позволяет не только сократить влияние случайных факторов на работу нейронной сети (как при инициализации весовых коэффициентов, так и при формировании обучающей выборки), но и увеличить количество используемых при обучении данных, что может быть особенно важным, когда количество данных ограничено.

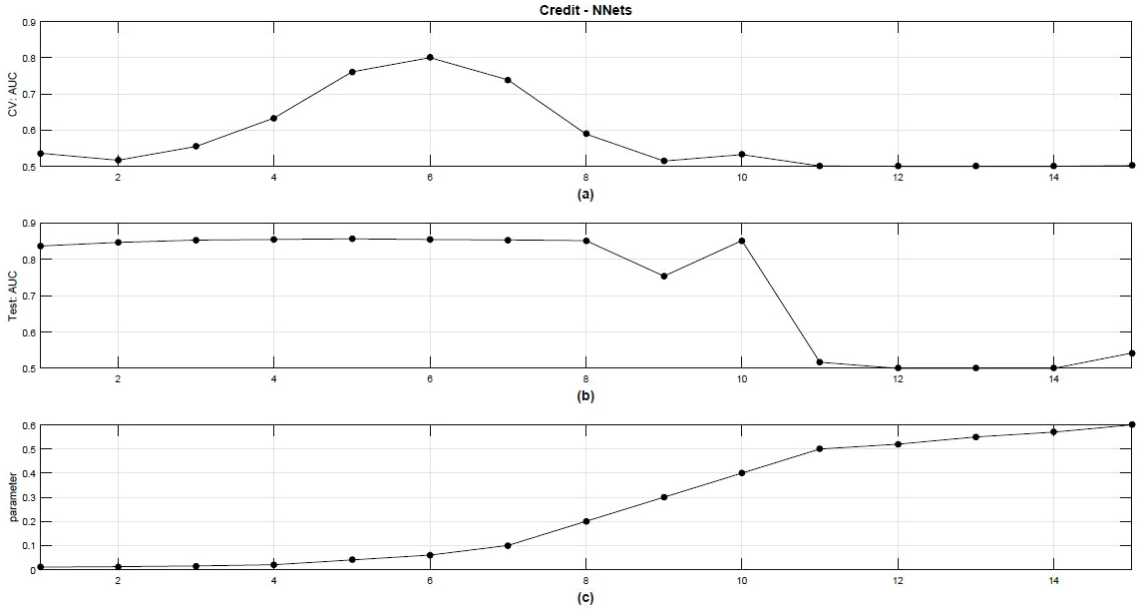

Рис. 1. Динамика качества классификации в модели $\{nnet, Credit\}$ в зависимости от выбора параметра β (1).

Обозначим отдельное случайное подмножество как $RS(\alpha, \beta)$, где α — пропорция «плохих», β — пропорция «хороших» клиентов. RS формируются по следующему правилу:

$$RS(\alpha, \beta) = \begin{cases} \xi \le \alpha, & \text{если y = 1} \\ \xi \le \beta, & \text{если y = 0} \end{cases}, \tag{1}$$

где ξ — непрерывная равномерно распределённая случайная величина на интервале $\{0, \ldots, 1\}$.

Важными параметрами данного алгоритма являются n и $q \leq 1$ — пропорциональное соотношение «плохих» и «хороших» клиентов (в локальном тренировочном подмножестве). Значения параметров определяются эмпирическим путём, но важно, чтобы n было достаточно большим, а q не слишком маленьким [3]. Например, параметры $\alpha = 0,85,\ \beta = 0,06,\ n = 20$, соответствуют 20-ти случайным подмножествам, каждое из которых включает 85% «плохих» и 6% «хороших» клиентов из исходной выборки.

Поскольку каждый отдельно взятый классификатор мы обучаем только на части данных, оставшиеся наблюдения могут быть использованы для проверки качества решения в соответствии с принципами скользящего контроля. Таким образом, локальные оценки будут накапливаться параллельно вычислению однородного ансамбля, а их арифметическое среднее мы назовём паспортом скользящего контроля, характеризующим качество финального решения [6].

В итоге структурная схема алгоритма может быть сформулирована следующим образом:

1. Инициализация параметров алгоритма α, β, n.
2. Для каждого i из $\{1, \ldots, n\}$
   (a) сформировать случайное сбалансированное подмножество $RS(\alpha, \beta)$ согласно правилу (1);
   (b) построить классификатор, используя $RS(\alpha, \beta)$ в качестве тренировочного множества;
   (c) получить прогноз для кросс-валидационной выборки (все наблюдения, не вошедшие в RS) и оценить качество полученного решения (см. рис. 1 a — иллюстрация усреднённых значений);
   (d) получить прогноз для валидационной выборки и оценить качество полученного решения (см. рис. 1 b — иллюстрация усреднённых значений).
3. Найти среднее арифметическое всех решений для тестовой выборки (финальное решение).

Рис. 1 с иллюстрирует последовательность значений параметра β. Оптимальное значение ($\beta = 0.6$) вполне соответствует нашим ожиданиям.

1.2.1. Результаты

В данном примере (без балансировки) $nnet$ модель производит нулевое решение (см. рис. 1). Это решение соответствует большим значениям параметра β. При значениях $\beta \leq 0,08$ качество решения существенно улучшается и достигает уровня $AUC = 0,8$.

2. ДАННЫЕ MNIST

Второй пример касается классических данных MNIST, в которых каждое наблюдение представляет собой матрицу 28×28. При загрузке в Matlab каждое отдельное значение матрицы определяет яркость и цвет. Каждая отдельная матрица соответствует определённой рукописной цифре. Задача состоит в распознавании этих цифр. Отметим, что второй пример имеет существенные отличия в сравнении с первым:

1) первая задача является бинарной, во второй задаче — 10 классов;

2) в первой задаче мы имеет только десять признаков, вторая задача является многомерной и включает 784 признака.

В случае многомерных данных хорошо себя зарекомендовал метод опорных векторов (svm). В программных средах R и Python имеются стандартные функции svm. Однако, прямое применение этих функций к данным MNIST производит очень плохие классификационные результаты. Качество классификации увеличивается существенным образом, если мы путём нормализации приведём исходные признаки к интервалу $[0, \ldots, 1]$.

Смешанный набор данных Национального института стандартов и технологий (Mixed National Institute of Standards and Technology, MNIST)[2] является к настоящему времени классическим и выполняет роль критерия, относительно которого тестируются различные методы распознавания образов и алгоритмы.

База данных MNIST включает в себя 42000 тренировочных образов и 28000 тестирующих изображений. Каждое изображение(рукописная цифра от нуля до девяти) представляет собой квадратную матрицу размерности 28. Значения элементов матрицы лежат в диапазоне от 0 (представляет белый цвет) до 255 (представляет черный цвет). Промежуточные значения отражают оттенки серого (см. рис. 2).

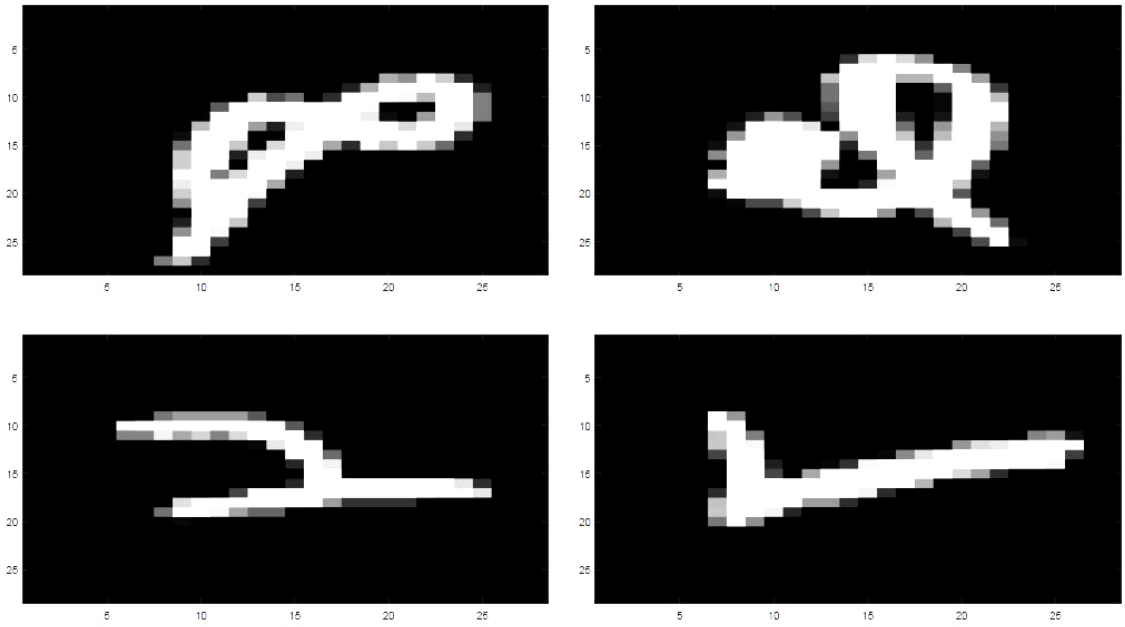

Рис. 2. Четыре иллюстрации изображений из базы данных MNIST

В качестве критерия качества алгоритма рассмотрим точность классификации, то есть пропорцию изображений, которые были правильно классифицированы. Например результат 0.97 означает то, что правильно распознаны все, за исключением 3% изображений.

2.1. Метод опорных векторов

Метод опорных векторов (Support Vector Machine, SVM) зарекомендовал себя очень эффективным при работе с данными больших размерностей (или имеющими большое

[2] http://yann.lecun.com/exdb/mnist.html

количество объясняющих переменных). В случае данных MNIST размерность $m = 784$. Этот алгоритм является одним из наиболее популярных методов машинного обучения с учителем. Целью метода опорных векторов является построение оптимальной разделяющей гиперплоскости в пространстве признаков большой размерности. Следует отметить, что метод опорных векторов позволяет добиться высокого качества распознавания цифр только при условии надлежащей нормализации входных признаков.

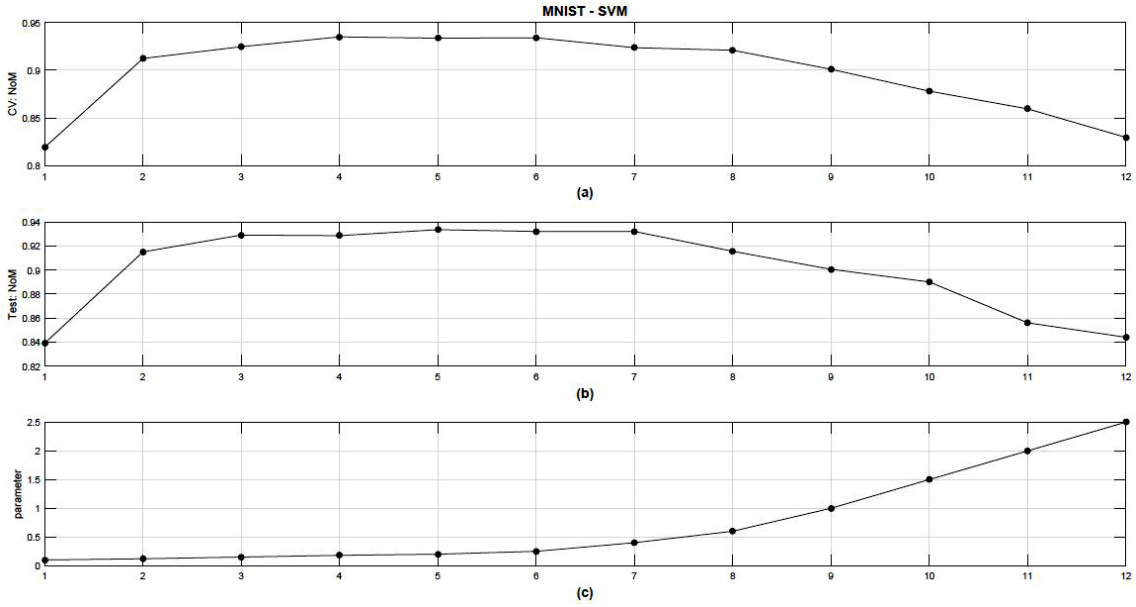

Рис. 3. Динамика качества классификации в модели $\{svm, MNIST\}$ в зависимости от выбора параметра γ (3).

2.2. Нормализации входных признаков

Следует отметить, что некоторые алгоритмы, такие как k-Means, Support Vector Machine и Non-Negative Matrix Factorization, «выигрывают» от нормализации признаков. Под нормализацией мы понимаем приведение значений признаков к определённому интервалу. Нормализация гарантирует, что признаки не получат искусственной надбавки, вызванной разницей в диапазонах, в которых лежат их значения.

Простейшая нормализация исходных матричных данных выполняется путем деления каждого значения из столбца на максимальный элемент из этого столбца $j = 1, \ldots, m$:

$$\phi = max(X[:, j]) + 0.001, \tag{2}$$

$$X[:, j] \leftarrow X[:, j] / (1.0 + \gamma \cdot \phi), \tag{3}$$

$$T[:, j] \leftarrow T[:, j] / (1.0 + \gamma \cdot \phi), \tag{4}$$

где $\gamma \geq 0$ — параметр нормализации.

Рис. 3 аналогичен рис. 1 и иллюстрирует (сверху вниз):

1. (a): результаты скользящего контроля;

2. (b): результаты тестирования относительно независимых данных;
3. (c): соответствующий параметр нормализации γ (2).

Согласно рис. 3(b) нормализация должна быть достаточно сильной, но не слишком сильной. Оптимальное значение $\gamma = 0.5$ соответствует максимальной пропорции верно классифицированных изображений.

2.2.1. Результаты

Качество классификации 0.84 — процент правильного распознавания (без нормализации) и 0.92 — после нормализации, см. рис. 3, где параметр γ выполняет роль регулятора силы нормализации.

3. ЗАКЛЮЧЕНИЕ

Интересно отметить, что во многих случаях прямое применение стандартных методов машинного обучения приводит к довольно плохим результатам. Однако ситуация может быть легко исправлена при помощи специальных «корректировочных» методов общего характера. Отметим, что выбор корректировочного метода зависит от каждого конкретного случая. В работе рассмотрены два примера в качестве иллюстраций: 1) модель $\{nnets, Credit\}$ — метод сбалансированных подмножеств в случае несбалансированных данных и 2) модель $\{svm, MNIST\}$ — нормализация входных признаков.

Список литературы

1. *Атхит М., Посов И.А.* Автоматизация проведения дистанционных соревнований, основанных на исследовательских сюжетах по математике и информатике // Компьютерные инструменты в образовании, 2014. № 6. С. 45–51.
2. *Дьяконов А.Г.* Алгоритмы для рекомендательной системы: технология LENKOR. // Бизнес-Информатика, 2012. Т. 1, № 19. С. 32–39.
3. *Никулин В.Н., Палешева С.А., Зубарева Д.С.* Об однородных ансамблях при использовании метода бустинга в приложении к классификации несбалансированных данных. // Вестник ПГУ. Серия: Экономика, 2012. С. 7–14.
4. *Lu Y., Guo H., Feldkamp L.* Robust neural learning from unbalanced data examples. // IEEE World Congress on Computational Intelligence, 1998. P. 1816–1821.
5. *Ciresan D., Meier U., Gambardella L., Schmidhuber J.* Deep Big Simple Neural Nets Excel on Handwritten Digit Recognition. // Neural Computation, Vol. 22(12). 2010. P. 3207—3220.
6. *Nikulin V. N.* On the Evaluation of the Homogeneous Ensembles with CV-passports. // PAKDD 2013, Springer LNCS 7867, J. Li et al. (Eds.) 2013. P. 109–120.

METHODS OF BALANCED RANDOM SETS AND DATA NORMALISATION FOR IMPROVEMENT OF CLASSIFICATION QUALITY

Nikulin V. N., Kanishchev I. S., Bagaev I. V.

Abstract

In many cases direct application of the standard classification models leads to poor quality of results. In this paper we consider two examples. The subject of the first example are popular imbalanced data "Credit" from the platform Kaggle. Standard function $nnet$ (neural networks) in the program environment R is used as a classificator. This function is ignoring an important minority class. As a solution to this problem, we are proposing to consider a large number of relatively small and balanced subsets, where elements were selected randomly from the training set. The subject of the second example are famous data MNIST and standard function svm (support vector machine) in the environment Python. The necessity of normalisation of the original features is demonstrated.

Keywords: *machine learning, data mining, neural networks, homogeneous ensemble, imbalanced data, patterns recognition, support vector machine.*

Vladimir Nikolaevich Nikulin, PhD, Associate Professor in Computer Science, Department of Mathematical Methods, Vyatka State University, Russia, 610000, Kirov, ul. Moskovskaya, 36
Email: vnikulin.uq@gmail.com

**Никулин Владимир Николаевич,
кандидат физико-математических наук,
доцент кафедры математических методов,
Вятский государственный университет,
Россия, 610000, Киров, ул. Московская, 36,
vnikulin.uq@gmail.com**

**Канищев Илья Сергеевич,
магистрант кафедры математических методов, Вятский государственный университет,
Россия, 610000, Киров, ул.Московская, 36,
kanishchev.ilya@gmail.com**

**Багаев Иван Владимирович,
магистрант кафедры математических методов, Вятский государственный университет, Россия, 610000, Киров, ул.Московская, 36,
iv.bagaew@yandex.ru**