

УДК 004.93:519.24

Нейросетевой механизм адаптации при решении кусочно- постоянной задачи анализа независимых компонент

Простов Ю.С.,* Тюменцев Ю.В.**

Московский авиационный институт (национальный исследовательский университет), МАИ, Волоколамское шоссе, 4, Москва, А-80, ГСП-3, 125993, Россия

**e-mail: prostov.yury@yandex.ru*

***e-mail: yutium@gmail.com*

Аннотация

Предлагается метод, который позволяет при определённых условиях применять неадаптивные нейросетевые модели для решения кусочно-постоянной задачи анализа независимых компонент. В основе метода лежит реализация механизма детектирования изменений в условиях решаемой задачи и автоматизация процесса переобучения нейронной сети непосредственно во время решения задачи. Кратко рассмотрен вопрос целесообразности использования предлагаемого механизма вместо чисто адаптивных нейросетевых моделей. На примере конкретной нейронной сети дано обоснование работоспособности метода и продемонстрированы результаты численного моделирования.

Ключевые слова: нейронные сети, адаптивность, анализ независимых компонент

Введение

Нейросетевые технологии всё чаще находят применение в различных областях науки и техники. Их универсальность позволяет решать такие важные для авиационной и ракетно-космической отрасли задачи, как распознавание и классификация образов, идентификация систем, аппроксимация функций, прогнозирование процессов и многое другое [1]. При этом применение нейросетевого подхода представляется особенно привлекательным в случаях, когда имеется возможность использовать системы с поддержкой параллельных вычислений.

Существующие нейросетевые модели можно разделить на два класса: неадаптивные и адаптивные модели. В данном случае под адаптивностью понимается наличие некоторого механизма, позволяющего корректировать параметры нейронной сети непосредственно во время решения задачи для более точного соответствия текущим условиям.

Очевидно, что для многих прикладных задач требуется применение именно адаптивных моделей. Однако это сопряжено с повышением требований к мощности используемых вычислительных систем, поэтому актуальна задача совершенствования таких моделей в целях снижения вычислительных затрат их функционирования с сохранением адаптивных свойств. Это позволит проводить более гибкое управление информационно-

вычислительными системами (ИВС) летательных и космических аппаратов, начиная от возможности решения дополнительных задач за счёт освободившихся ресурсов и заканчивая использованием менее мощных, но более привлекательных с точки зрения веса и энергозатрат ИВС.

Характерной чертой семейства неадаптивных нейросетевых моделей является то, что их параметры полностью определяются во время предварительного обучения [1], в то время как в адаптивных моделях, как правило, необходимо непрерывно вычислять статистики высокого порядка [2]. В результате, применение адаптивных нейронных сетей приводит к резкому росту числа выполняемых матричных операций, а значит, как было сказано ранее, к повышению требований к вычислительным системам. Кроме того, для таких сетей остаётся открытым вопрос сходимости, а значит и адекватности получаемого решения. Поэтому во многих случаях, если есть возможность, используют неадаптивные модели, накладывая ограничения на задачу и принимая различные допущения. Это позволяет получать хоть и приближённые, но приемлемые решения с более низкими вычислительными затратами.

В данной статье предлагается метод, который позволяет преобразовывать неадаптивные нейросетевые модели в адаптивные, используя принцип «переобучение по мере необходимости». Его суть заключается в том, что если в процессе решения задачи имеется возможность

установить факт изменения текущих условий, то можно попытаться налету переобучить применяемую неадаптивную нейронную сеть для соответствия новым условиям непосредственно в процессе её функционирования. Для этого необходимо, во-первых, чтобы задача обладала свойством кусочного постоянства упомянутых условий и для неё существовала метрика, отражающая наличие изменений в задаче, и, во-вторых, чтобы алгоритм обучения с соответствующими корректировками мог бы быть применён после периода предварительного обучения. Если для задачи и используемой нейронной сети эти условия выполнимы, то неадаптивную нейросетевую модель с некоторыми оговорками можно преобразовать в адаптивную.

Для апробации предлагаемого механизма рассматривается задача анализа независимых компонент (АНК), которая позволяет решать такие задачи, как слепое разделение сигналов и снижение размерности данных [3]. Для АНК характерно наличие как адаптивных [2], так и неадаптивных [4] нейросетевых методов решения, которые имеют упомянутые ранее достоинства и недостатки. Результатом работы стало успешное преобразование неадаптивной нейронной сети в адаптивную с демонстрацией её работоспособности на примере решения задачи АНК.

Анализ независимых компонент

Рассмотрим задачу АНК с постановки её стационарного варианта. Пусть дана следующая статистическая модель [2]:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + e, \quad (1)$$

где $\mathbf{x} \in \mathbf{R}^N$ и $\mathbf{s} \in \mathbf{R}^M$ – случайные векторы, \mathbf{A} – обратимая вещественная матрица размерности $N \times M$ и $e \in \mathbf{R}^N$ – случайный вектор, отражающий наличие шума в модели. Тогда решение задачи заключается в том, что по наблюдениям за вектором \mathbf{x} необходимо найти матрицу \mathbf{A} и оценить случайный вектор \mathbf{s} при условии, что:

1. Компоненты вектора \mathbf{s} статистически независимы:

$$\forall i, j \quad P(s_i s_j) = P(s_i)P(s_j).$$

2. Компоненты вектора \mathbf{s} имеют распределение, отличное от нормального (допускается, что только одна из компонент распределена нормально).
3. Все случайные величины центрированы: $E[\mathbf{x}] = 0$, $E[\mathbf{s}] = 0$, $E[e] = 0$.

Таким образом, множество независимых компонент $\{s_i\}_{i=1}^M$ представляет собой набор независимых относительно друг друга сигналов, линейная комбинация которых отражает компоненты входного сигнала, т.е.

$$\forall i \quad x_i = \sum_{j=1}^M a_{ij} s_j.$$
 Для разложения входного сигнала на независимые сигналы

используется матрица \mathbf{W} , обратная матрице \mathbf{A} : $\mathbf{s} = \mathbf{W}\mathbf{x}$.

В нестационарном варианте задачи предполагается, что набор НК зависит от времени, в результате чего выражение (1) приобретает следующий вид:

$$\mathbf{x}(t) = A\mathbf{s}(t) + e. \quad (2)$$

Далее в работе мы будем рассматривать частный случай задачи (2), в которой предполагается существование кусочно-постоянного решения. То есть подразумевается, что весь временной интервал t^{all} , на котором рассматривается задача, можно разбить на непересекающиеся промежутки Δt_k : $\bigcup_k \Delta t_k = t^{all}$ и $\bigcap_k \Delta t_k = \emptyset$. При этом на каждом из них задача будет стационарной и решения, полученные на соседних интервалах, различны:

$$\begin{cases} s(t_i) = s(t_j) & , \text{если } t_i, t_j \in \Delta t_k \\ s(t_i) \neq s(t_j) & , \text{если } t_i \in \Delta t_k \text{ и } t_j \in \Delta t_{k+1} \end{cases} \quad (3)$$

Нейросетевое решение

В настоящее время существует целый ряд нейросетевых моделей, которые успешно применяются для решения стационарной задачи АНК. Для демонстрации работоспособности предлагаемого механизма была использована искусственная нейронная сеть, предложенная в работе [5], так как она способна решать стационарную задачу АНК, что продемонстрировано в работе [4], и при этом достаточно проста в понимании и реализации.

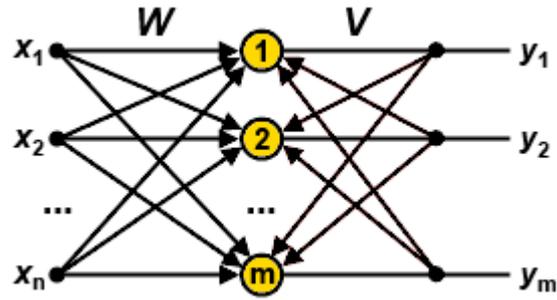


Рис. 1. Архитектура рассматриваемой искусственной нейронной сети

Как показано на рис. 1, выбранная нейронная сеть представляет собой однослойную сеть прямого распространения с латеральными тормозными связями, нейроны которой преобразуют входной вектор $x \in \mathbf{R}^N$ в выходной вектор $y \in \{0;1\}^M$ согласно пороговой функции активации:

$$y = \theta(u - 0,5), \quad (4)$$

где $\theta(s)$ – пороговая функция Хэвисайда, а вектор $u \in \mathbf{R}^M$ отражает внутреннее состояние нейронов и при предъявлении очередного входного вектора x вычисляется согласно следующему выражению:

$$\begin{cases} \frac{du}{d\tau} = \frac{1}{1 + e^{-\lambda(Wx - Vu - r)}} - u, \\ u(0) = Wx - r \end{cases} \quad (5)$$

где $W \in \mathbf{R}^{M \times N}$ – весовая матрица, соответствующая входным возбуждающим связям, $V \in \mathbf{R}^{M \times M}$ – весовая матрица декорреляции, соответствующая латеральным тормозным связям, а $r \in \mathbf{R}^M$ – пороговый вектор. Переменные

W , V и r вычисляются в ходе обучения сети, а параметр λ подбирается вручную и влияет на скорость сходимости решения.

Предварительное обучение нейронной сети представляет собой итеративный процесс: на каждой итерации вектору x присваивается некоторое значение, выбираемое в случайном порядке из заранее составленной выборки X , после чего корректируются переменные W , V и r . При этом выборка X должна качественно отражать входной сигнал, который предстоит обрабатывать сети в процессе непосредственного применения. Как правило, от размера выборки зависит затраченное на обучение количество итераций, которое называют временем обучения T .

В соответствии с оригинальной работой [5], на каждой итерации для изменения матрицы W применяют хеббовское правило обучения, а для матрицы V – анти-хеббовское правило:

$$\begin{cases} \Delta w_{ij} = \alpha y_i (x_j - w_{ij}) \\ \Delta v_{ij} = \beta (y_i y_j - \mu^2), \\ \Delta r_i = \gamma (y_i - \mu) \end{cases} \quad (6)$$

где параметр μ – отражает априорную вероятность появления образа, а переменные α , β и γ определяют так называемую скорость обучения связей, то есть задают степень изменения весов на каждой итерации. Важно заметить, что эти скорости не фиксированы и линейно убывают

относительно произвольных начальных значений α^0 , β^0 , γ^0 в течение всего периода обучения:

$$\alpha(t) = \alpha^0 * (1-t/T), \quad (7)$$

аналогично для β и γ .

Также оригинальная модель предполагает, что на элементы матрицы V накладывается условие: $v_{ij} = 0$, когда $i = j$ или $v_{ij} < 0$. В работе [4] показано, что для ускорения процесса сходимости на матрицу W можно наложить дополнительное ограничение: $w_{ij} = 0$, когда $w_{ij} < 0$.

В результате, после завершения обучения мы получим решение в виде весовых матриц W и V , а также порогового вектора r . В дальнейшем их можно применить для разложения входного сигнала x на независимые компоненты согласно формуле (4), в которой наличием НК s_i соответствует ненулевое значение переменной y_i .

Стоит отметить, что описанная модель использует довольно сильное допущение в правиле обучения (6): каждая из НК должна появляться во входном сигнале с заданной априорной вероятностью μ . Однако в предлагаемом методе, как будет показано далее, не делается каких-либо явных предположений относительно способа изменения связей, за

исключением наличия скоростей их изменения. Поэтому такое допущение приемлемо и не влияет на основной результат работы.

Механизм адаптации

Основная идея предлагаемого метода заключается в том, что в процессе применения нейронной сети мы можем попытаться отслеживать изменения в условиях решаемой задачи, наблюдая за динамикой нейронной сети. Если какие-либо изменения будут обнаружены, то сеть можно попробовать переобучить.

В простейшем случае, т.е. при решении задачи АНК выбранной моделью, каждому нейрону y_i соответствует некоторая НК s_i , что позволяет проводить анализ динамики каждого нейрона независимо от других и делать выводы относительно изменений соответствующей НК. Кроме того, используя условие кусочной постоянности, такой анализ может быть основан на оценивании активности нейрона $\langle y_i \rangle$, т.е. на усреднённом по времени значении переменной y_i .

Можно заметить, что вследствие бинарной природы вектора y активность нейрона отражает оценку вероятности \hat{p}_i появления НК s_i во входном сигнале: $\hat{p}_i = P(s_i) = P(y_i = 1) = \lim_{t \rightarrow \infty} \langle y_i \rangle$. Кроме того, из (4) и (5) следует, что $y_i = 1$, когда:

$$w_i x > r_i + v_i y. \quad (8)$$

Следовательно:

$$P(y_i = 1 | x) = p(w_i x > r_i + v_i y | x) = \hat{p}_i. \quad (9)$$

Без потери общности можно предположить, что w_i и x нормированы, тогда величина $w_i x$ отражает скалярное произведение входного вектора x и значения i -ой НК w_i нейронной сети. Поэтому, если при изменении условий задачи среди нового набора независимых компонент не окажется НК, во-первых, достаточно близкой вектору w_i , чтобы выполнялось неравенство (8), и, во-вторых, чьё распределение вероятности будет сохранять равенство (9), то активность i -ого нейрона изменится. Следовательно, необходимо переобучение модели.

Также отклонения в активности i -ого нейрона могут быть связаны с правой частью выражения (8): если при изменении условий задачи, i -ая НК остаётся прежней, но меняются распределения других НК, то равенство (9) может быть нарушено. В таком случае необходимо переобучение сети для корректировки параметров модели.

Таким образом, если в процессе функционирования нейронной сети колебания величины $\langle y_i \rangle$ превышают некоторые допустимые пределы δ , связанные с погрешностями вычислений и наличием шума, то мы можем

предположить наличие изменений в условиях решаемой задачи, и сеть требуется переобучить.

Однако, в отличие от процесса предварительного обучения, время обучения T здесь не определено, поэтому правило (7) необходимо модифицировать. Если предположить, что независимые компоненты представляют собой пуассоновский процесс, то изменения могут иметь следующий вид:

$$\begin{cases} \Delta \langle y_i \rangle = h(\langle y_i \rangle) - \langle y_i \rangle \\ \alpha_i = g_\delta(\Delta \langle y_i \rangle) \end{cases}, \quad (10)$$

где функция h является функцией запаздывания и позволяет оценить колебания активности нейрона $\Delta \langle y_i \rangle$, а функция g преобразует эту величину в коэффициент изменения связей, например следующим образом:

$$g_\delta(s) = \begin{cases} \alpha * |s| & , \text{если } |s| > \delta \\ 0 & , \text{если } |s| \leq \delta \end{cases}. \quad (11)$$

Аналогично проводятся вычисления для β и γ .

Таким образом, заменяя линейно убывающие коэффициенты скорости обучения (7) на правило (8), зависящее от активности нейронов, мы можем отслеживать изменения в условиях задачи непосредственно во время функционирования нейронной сети и переобучать её, когда такие изменения обнаружены. Другими словами, такая замена позволяет нам преобразовать

неадаптивную нейросетевую модель в адаптивную при соблюдении отмеченных ранее условий, а использование различных функций h и g позволяет контролировать чувствительность и инерционность модели, то есть менять период, в течение которого нейроны будут сохранять текущие значения весовых коэффициентов и не переобучаться.

При этом нельзя не упомянуть о проблеме сходимости полученной модели в том смысле, что интервалы постоянства условий задачи Δt_k могут быть меньше времени, необходимого для переобучения сети. Очевидно, что в таком случае получившаяся адаптивная модель будет непригодной для применения. Однако невозможно заранее установить минимально допустимый размер интервала Δt_k , так как для рассматриваемой нейронной сети, как и для большинства других моделей, не представляется возможным однозначно установить время, необходимое для сходимости решения. В лучшем случае, можно утверждать только о сходимости за конечное число шагов [1]. Как правило, данную проблему пытаются решать за счёт проведения тестовых запусков, во время которых подстраивают параметры нейронной сети для достижения наилучшего результата, а также за счёт использования опыта исследователя и априорных знаний о предметной области.

Моделирование

Для проверки работоспособности предлагаемого механизма была использована довольно распространённая в области АНК задача «Földiák bars» [5]. В этой задаче входные данные представляют собой бинарные изображения размером 8 на 8 пикселей. При генерации изображений (см. рис. 2) допускается использование 8 горизонтальных и 8 вертикальных линий (см. рис. 3), каждая из которых появляется на изображении с вероятностью $1/8$ независимо от остальных. При этом не обязательно, чтобы были использованы все 16 линий. Таким образом, каждая из полос представляет собой независимую компоненту, и решение заключается в детектировании множества линий, использованных при генерации изображений.

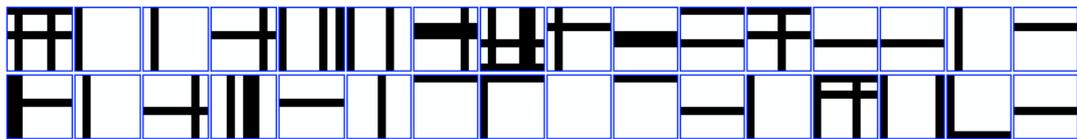


Рис. 2. Примеры используемых изображений



Рис. 3. Линии, формирующие входные изображения

В процессе численного моделирования была использована сеть с 16 нейронами (в соответствии с максимально возможным числом независимых компонент) и следующими параметрами: $\mu = 1/8$, $\alpha^0 = 0.1$, $\beta^0 = 0.005$, $\gamma^0 = 0.1$,

$\lambda = 20$, $T = 12000$. В начале обучения вектор r и матрица V обнулялись, а элементам матрицы W присваивались случайные значения, близкие к нулю.

Первый этап моделирования заключался в обучении оригинальной модели. При этом было задействовано правило линейного убывания параметров (7) с использованием заранее заготовленной выборки изображений, включающей 256 уникальных паттернов. В формировании паттернов было задействовано 12 из 16 НК, изображённых на рис. 4 (а). В результате, к концу периода обучения задача АНК была успешно решена, что отражено на рис. 5.

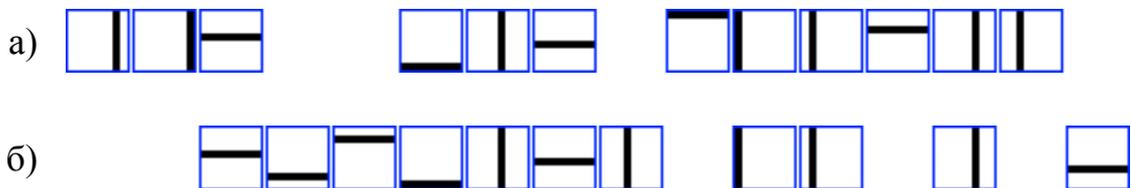


Рис. 4. Независимые компоненты, использованные: а) во время обучения и проверки работоспособности оригинальной модели; б) после проверки работоспособности оригинальной модели

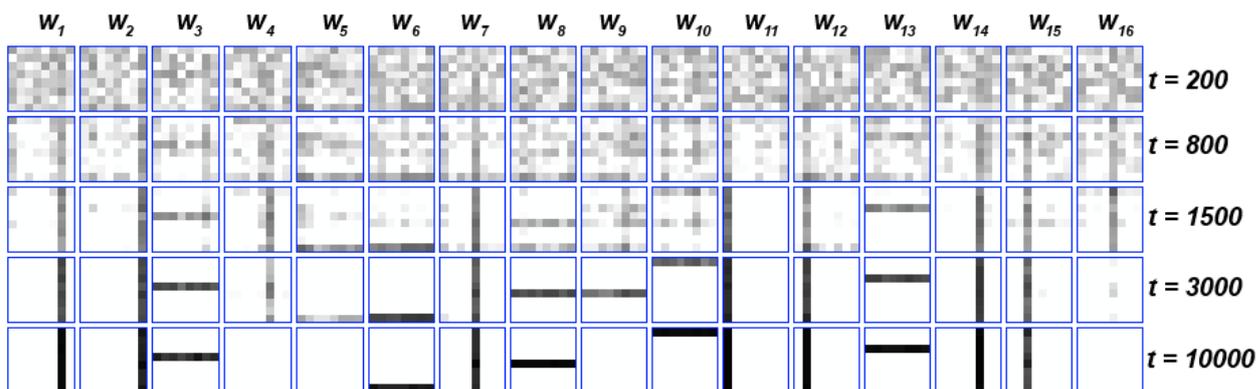


Рис. 5. Изменение коэффициентов векторов-строк весовой матрицы W во время обучения сети (каждый вектор трансформирован в матрицу)

Следующий этап заключался в проверке работоспособности обученной модели. Для этого на вход сети подавались изображения, сгенерированные из тех же НК, но не входящие в состав обучающей выборки. Нейронная сеть успешно детектировала наличие использованных в изображении линий. Однако далее, в момент времени $t = 15000$, набор независимых компонент был изменён: новый включал 11 из 16 компонент, изображённых на рис. 4 (б). На рис. 6 показано усреднённое значение активностей нейронов сети. Видно, что изменения во входных данных привели к снижению этой величины. Это вызвано тем, что часть нейронов, соответствующая пропавшим компонентам, перестала активироваться ($y_i = 0$). В тоже время наличие новых линий на изображении не детектируется, так как сеть не была обучена для их обнаружения.



Рис. 6. Среднее значение активности элементов сети

Третий этап подразумевал использование предложенного адаптивного механизма. Для этого были повторно выполнены первые два этапа, но по окончании периода обучения было задействовано правило (8). В результате, нейросетевая модель смогла адаптироваться к произошедшим в момент времени $t = 15000$ изменениям, что отражено на рис. 7: отсутствующие компоненты больше не отражены в структуре сети, а новые были успешно обнаружены.

Однако стоит отметить, что одна из новых НК не была найдена. Это связано с тем, что реальная частота её появления значительно меньше ожидаемой ($\hat{p}_i = 0.101 \ll \mu = 0.125$), в результате чего нейронная сеть просто не способна зафиксировать её наличие во входном сигнале даже в случае использования оригинального алгоритма (7) при сколь угодно большом времени обучения. Поэтому можно утверждать, что данный факт не связан с использованием предлагаемого метода.

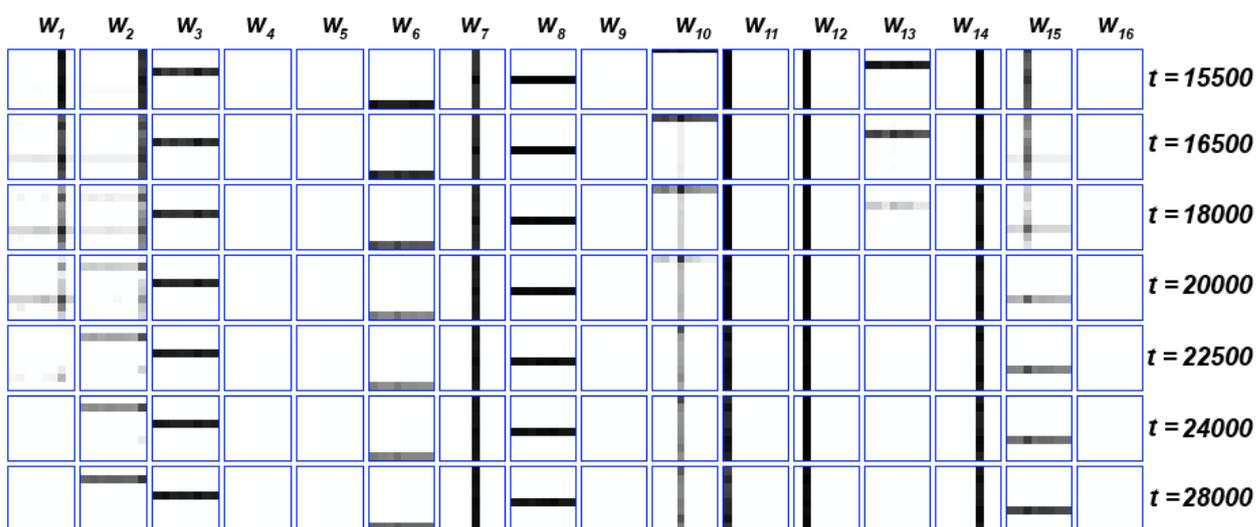


Рис. 7. Изменение коэффициентов векторов-строк весовой матрицы W во время адаптации сети (каждый вектор трансформирован в матрицу)

Заключение

Таким образом, выбранная нейронная сеть была успешно преобразована в адаптивную модель и применена для решения кусочно-постоянной задачи АНК. При этом не потребовалось вносить каких-либо серьёзных изменений в оригинальную модель, так как представленный механизм затрагивает только параметры скорости обучения связей.

Принципиальная возможность разделения представленного механизма на две части (обнаружение изменений во входном сигнале и переобучение нейронной сети) позволяет в значительной мере экономить вычислительные ресурсы, так как во время функционирования системы процесс детектирования изменений вносит небольшие дополнительные затраты, но

позволяет выполнять трудоёмкое переобучение сети только по мере необходимости.

Такой подход позволяет в некоторых случаях отказаться от применения чисто адаптивных нейросетевых моделей в пользу представленного метода, сводя к минимуму вычислительные затраты в периоды постоянства условий решаемой задачи. Для систем, функционирующих в режиме реального времени, это является важным фактором. Однако требуется дополнительный анализ для определения семейства нейросетевых моделей и класса задач, для которых возможно применение предлагаемого метода.

Кроме того, использование данного механизма накладывает ограничение на входной сигнал: предполагается, что НК в периоды постоянства должны обладать периодичностью или, в более общем случае, представлять собой пуассоновский процесс для того, чтобы колебания активности нейронов лежали в допустимых пределах. Возможность усовершенствования метода и, как следствие, ослабления данного условия также требует дальнейших исследований.

Библиографический список

1. Хайкин С. Нейронные сети: Пер. с англ. – М.: ИД «Вильямс», 2006. – 1104 с.
2. Choi S., Cichocki A., Amari S. Equivariant nonstationary source separation. *Neural Networks*, 2002, Vol. 15, pp. 121–130.

3. Hyvärinen A., Karhunen J., Oja E. Independent component analysis, New York, John Wiley, 2001, 505 p.
4. *Fyfe C. Hebbian learning and negative feedback networks (Advanced Information and Knowledge Processing Series)*, London, Springer, 2005, 383 p.
5. Földiák P. Forming sparse representations by local anti-Hebbian learning. *Biological Cybernetics*, Cambridge, 1990, vol. 64, no. 2, pp. 165–170.