Министерство образования и науки Российской Федерации
Московский физико-технический институт (государственный университет)
Факультет управления и прикладной математики
Вычислительный центр им. А. А. Дородницына РАН
Кафедра «Интеллектуальные системы»

Попова Мария Сергеевна

Выбор оптимальной сети глубокого обучения в задаче классификации временных рядов

010900 — Прикладные математика и физика

Бакалаврская диссертация

Научный руководитель:
ст.н.с ВЦ РАН, д.ф.-м.н.
Стрижов Вадим Викторович

Москва
2015 г.

Содержание

1 Введение

Диссертационная работа посвящена проблеме выбора оптимальной модели для многоклассовой классификации временных рядов акселерометра.

Актуальность темы

Для получения точного и устойчивого прогноза физической активности человека необходимы методы, позволяющие выбирать адекватные модели из некоторого множества допустимых моделей-претендентов. Настройка параметров универсальной модели является нетривиальной многоэкстремальной оптимизационной задачей. Предлагается упростить эту задачу, рассматривая наборы последовательно порождаемых устойчивых моделей заданной сложности. Модели порождаются путем модификации структуры искусственной нейронной сети глубокого обучения.Решается задача последовательной модификации нейронных сетей глубокого обучения. Требуется получить модель с небольшим числом связей между нейронами, которая достаточно точно решала бы задачу классификации физической активности человека по показаниям акселерометра и обладала бы устойчивостью к возмущениям данных. Ввиду этого возникает задача минимизации сложности модели без потери точности классификации.

В данной работе предлагается стратегия пошаговой модификации нейронной сети, комбинирующая этапы добавления и удаления параметров. Процедура модификации начинается с модели субоптимальной сложности и на каждом шаге добавляет в модель или удаляет из нее один параметр. Предлагается рассматривать такую процедуру пошаговой модификации модели как путь в многомерном кубе.

В вычислительном эксперименте предложенная стратегия тестируется на прикладной задаче. В качестве тестового примера рассматривается задача классификации физической активности человека по измерениям акселерометра.

Целью работы является исследование проблемы выбора оптимальных моделей для многоклассовой классификаци временных рядов.

Основные положения, выносимые на защиту:

1. Предложена стратегия пошаговой модицикации модели, которая позволяет получать точные и усточивые нейронные сети с небольшим числом параметров.

2. Предложена суперпозиция моделей для выделения информативных признаков из временных рядов.

3. Проведен вычислительный эксперимент на двух наборах данных. Результаты сравнивались с работами зарубежных авторов.

Научная новизна:

1. Предложена стратегия порождения точных и устойчивых моделей оптимальной сложности;

2. Предложена суперпозиция нейронных сетей глубокого обучения для выделения информативных признаков из временных рядов.

Практическая значимость

Предлагаемая в работе стратегия порождения моделей предназначена непосредственно для применения на практике. Модели, порождаемые предложенной стратегией могут использоваться для решения задач классификации типов физической активности человека по измерениям с датчиков мобильного устройства.

Достоверность изложенных в работе результатов подтверждена математическими доказательствами, экспериментальной проверкой полученных алгоритмов на реальных задачах регрессии; публикациями результатов исследования в рецензируемых научных изданиях, в том числе рекомендованных ВАК.

Апробация работы. Основные результаты работы докладывались на:

- 57-я международная научная конференция МФТИ (Москва-Долгопрудный-Жуковский, 2014);

- Конференция студентов, аспирантов и молодых ученых "Ломоносов-2015"(Москва, 2015).

Публикации. Основные результаты по теме диссертации изложены в 2 печатных изданиях, 1 из которых издано в журнале, рекомендованном ВАК, 1 — в тезисах докладов.

1. Попова М.С., Стрижов В.В. Выбор оптимальной модели классификации по измерениям акселерометра // Информатика и ее применения. Т. 9. Вып. 1. С. 79-89

2. Попова М.С., Стрижов В.В. Построение нейронных сетей глубокого обучения для классификации временных рядов // Системы и средства информатики (принято к публикации).

2 Постановка задачи выбора оптимальной модели классификации

2.1 Постановка задачи

В данной работе решается проблема построения оптимальных устойчивых моделей в задаче классификации физической активности человека. Каждый тип физической активности конкретного человека описывается набором признаков, сгенерированных по временным рядам с акселерометра. В условиях мультиколлинеарности признаков выбор устойчивых моделей классификации затруднен из-за необходимости оценки большого числа параметров этих моделей. Оценка оптимального значения параметров также затруднена в связи с тем, что функция ошибок имеет большое количество локальных минимумов в пространстве параметров. В работе исследуются модели, принадлежащие классу двуслойных нейронных сетей. Ставится задача нахождения Парето-оптимального фронта на множестве допустимых моделей. Предлагаются критерии оптимального, последовательного и устойчивого прореживания нейронной сети, критерий наращивания сети, а также строится стратегия пошаговой модификации модели с использованием предложенных критериев. В вычислительном эксперименте модели, порождаемые предложенной стратегией, сравниваются по трем критериям качества — сложности, точности и устойчивости.

Для получения точного и устойчивого прогноза физической активности человека необходимы методы, позволяющие выбирать адекватные модели из некоторого множества допустимых моделей-претендентов. Проблема выбора моделей обсуждается в работах [1–3]. Настройка параметров универсальной модели является нетривиальной многоэкстремальной оптимизационной задачей. Предлагается упростить эту задачу, рассматривая наборы последовательно порождаемых устойчивых моделей заданной сложности. Модели порождаются путем модификации структуры искусственной нейронной сети. Решается задача последовательной модификации нейронной сети. Требуется получить нейронную сеть с небольшим числом связей между нейронами, которая достаточно точно решала бы задачу классификации физической активности человека по показаниям акселерометра и обладала бы устойчивостью к возмущениям данных. Ввиду этого возникает задача минимизации сложности модели без потери точности классификации [4].

Существует два базовых подхода к решению задачи выбора сетей оптималь-

ной структуры: *наращивание структуры сети* (network growing) [5] и *прореживание структуры сети* (network pruning) [6–8].

Согласно первому подходу в качестве начальной модели выбирается сеть недостаточной сложности, решающая поставленную задачу с большим значением функции ошибки, после чего в сеть добавляются новые нейроны и связи между ними. В [5] описаны некоторые методы наращивания, приведен сравнительный анализ генетических алгоритмов с алгоритмом байесовской оптимизации. В алгоритмах метода прореживания модифицируется многослойная сеть с избыточным числом нейронов и связей между ними. Классическими алгоритмами прореживания нейронных сетей являются «optimal brain damage» [7] и «optimal brain surgery» [8], основанные на вычислении вторых производных функции ошибки. Также получили развитие *гибридные алгоритмы*, в которых объединяются оба упомянутых выше подхода [9–11].

В данной работе предлагается стратегия пошаговой модификации нейронной сети, комбинирующая этапы добавления и удаления параметров [12, 14]. Стратегия включает в себя критерии прореживания и наращивания структуры сети, критерии останова этапов добавления и удаления параметров, а также критерий останова процедуры модификации. Согласно предложенной стратегии процедура модификации начинается с нейронной сети избыточной сложности и чередует шаги удаления и добавления параметров до тех пор, пока этот процесс не стабилизируется согласно критерию останова процедуры модификации. Критерии прореживания и наращивания позволяют на каждом шаге процедуры модификации выбирать параметр, добавление или удаление которого улучшит качество нейронной сети, которое оценивается по трем критериям качества — сложности, точности и устойчивости [3, 13, 15]. Также предлагается рассматривать процедуру пошаговой модификации нейронной сети как путь в многомерном кубе.

В вычислительном эксперименте определяются значения критериев качества для нейронных сетей, порождаемых предложенной стратегией. В качестве тестового примера рассматривается задача классификации физической активности человека по измерениям акселерометра [36].

Дана выборка $\mathfrak{D} = \{(\mathbf{x}_i, \mathbf{t}_i)\}, i \in \mathcal{I} = \{1 \dots m\}$, состоящая из m объектов $\mathbf{x}$, каждый из которых описывается n признаками, $\mathbf{x}_i \in \mathbb{R}^n$, и принадлежит одному из z классов $\mathbf{t}_i \in \{0, 1\}^z$. Также задано разбиение множества индексов выборки $\mathcal{I} = \mathcal{L} \sqcup \mathcal{T}$ на обучающую $(\mathbf{x}_i, \mathbf{t}_i), i \in \mathcal{L}$, и контрольную $(\mathbf{x}_i, \mathbf{t}_i), i \in \mathcal{T}$. Требуется выбрать устойчивую модель классификации оптимальной сложности.

Определение 2.1. *Моделью назовем отображение*

$$\mathbf{f} : (\underset{k\times 1}{\mathbf{w}}, \underset{1\times n}{\mathbf{x}}) \mapsto \underset{1\times z}{\mathbf{y}},$$

где $\mathbf{w} = [w_1, \ldots, w_j, \ldots, w_k]^\mathsf{T}, j \in \mathcal{J} = \{1, \ldots, k\}$, *— вектор параметров модели,* $\mathbf{X} \in \mathbb{R}^{n\times m}$ *— матрица плана,* $\mathbf{y} \in \{0, 1\}^z$ *— зависимая переменная.*

Предполагается, что переменная $\mathbf{y}$ — мультиномиально распределенная случайная величина, а переменная $\mathbf{w}$ имеет нормальное распределение с нулевым математическим ожиданием:

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{A}^{-1}), \tag{2.1}$$

где $\mathbf{A}^{-1}$ — ковариационная матрица параметров общего вида, положительно-определенная: $\mathbf{w}^\mathsf{T}\mathbf{A}\mathbf{w} > \mathbf{0}$ для любого $\mathbf{w} \in \mathbb{R}^k$.

В данной работе рассматриваются модели, принадлежащие классу двухслойных нейронных сетей с функциями активации tanh и softmax:

$$\mathbf{a}(\mathbf{x}) = \underset{N_h\times z}{\mathbf{W}_2^\mathsf{T}}\, \mathbf{tanh}(\underset{n\times N_h}{\mathbf{W}_1^\mathsf{T}}\, \mathbf{x}), \tag{2.2}$$

$$\mathbf{f}(\mathbf{x}) = \frac{\exp(\mathbf{a}(\mathbf{x}))}{\sum_{j=1}^{n}\exp(a_j(\mathbf{x}))}.$$

Вектор $\mathbf{f}$ интерпретируется как вектор вероятностей: f_ξ есть вероятность того, что вектор $\mathbf{x}$ принадлежит классу с номером ξ:

$$\mathbf{f}(\mathbf{x}) = \{f_\xi\}, \quad 0 \leqslant f_\xi \leqslant 1, \quad \sum f_\xi = 1, \quad \xi = 1 \ldots z.$$

Под вектором параметров двухслойной нейронной сети будем понимать $\mathbf{w} = \mathrm{vec}(\mathbf{W}_1^\mathsf{T}|\mathbf{W}_2^\mathsf{T})$, где $\mathbf{W}_1, \mathbf{W}_2$ — матрицы весов первого и второго слоя нейронной сети (6.9). Вектор $\mathbf{y} = [y_1, \ldots, y_\xi, \ldots, y_z]^\mathsf{T}$ определим следующим образом:

$$y_\xi = \begin{cases} 1, & \text{если } \xi = \underset{\xi\in\{1,\ldots,z\}}{\operatorname{argmax}}(p_\xi), \\ 0, & \text{иначе.} \end{cases}$$

Вектор $\mathbf{y}$ — это вектор метки класса, полученный для объекта $\mathbf{x}$ с помощью построенной модели, в то время как вектор $\mathbf{t}$ – это вектор метки класса объекта $\mathbf{x}$ из выборки $\mathfrak{D}$.

Под *структурными* параметрами двухслойной нейронной сети будем понимать количество нейронов в скрытом слое нейронной сети — N_h. Матрица весов первого слоя

имеет размерность $n \times N_h$, матрица весов второго слоя имеет размерность $N_h \times z$. Далее будем считать, что структурные параметры фиксированы и одинаковы для всех рассматриваемых моделей.

Определение 2.2. *Параметр w_j модели $\mathbf{f}$ назовем активным, если $w_j \neq 0$.*

Определение 2.3. *Структурой $\mathcal{A}$ модели $\mathbf{f}$ назовем множество индексов активных параметров этой модели $\mathcal{A} = \{j : w_j \neq 0\} \subseteq \mathcal{J}$.*

Каждая структура $\mathcal{A} \subseteq \mathcal{J}$ задает некоторую модель

$$\mathbf{f}_\mathcal{A} : \hat{\mathbf{w}}_\mathcal{A} \in \mathbb{R}^k,$$

где $\mathbf{f}_\mathcal{A}$ — модель со структурой $\mathcal{A}$, а $\hat{\mathbf{w}}_\mathcal{A} \in \mathbb{R}^k$ — оптимальный вектор параметров модели $\mathbf{f}_\mathcal{A}$, определение которому будет дано ниже. Объединение всех $\mathbf{f}_\mathcal{A}$ назовем множеством допустимых моделей

$$\mathfrak{F} = \bigcup_{\mathcal{A} \subseteq \mathcal{J}} \{\mathbf{f}_\mathcal{A}\}. \tag{2.3}$$

Оптимальную модель $\hat{\mathbf{f}}_\mathcal{A}$ будем выбирать из множества допустимых моделей $\mathfrak{F}$.

Согласно гипотезе (5.1) о распределении многомерных случайных величин $\mathbf{y}$ и $\mathbf{w}$ в качестве функции ошибки выберем функцию

$$S(\mathbf{w}|\mathcal{K}) = -\sum_{i \in \mathcal{K}} \sum_{\xi=1}^{z} t_{i\xi} \ln\bigl(f_\xi(\mathbf{x}_i, \mathbf{w})\bigr), \tag{2.4}$$

максимизирующую логарифм правдоподобия случайной величины $\mathbf{y}$ и заданную на разбиении выборки $\mathfrak{D}$, определенном некоторым множеством индексов $\mathcal{K} \subseteq \mathcal{I}$, $\mathbf{t}_i = [t_{i1}, \ldots, t_{i\xi}, \ldots, t_{iz}]^\intercal$.

Определение 2.4. *Оптимальным вектором параметров модели $\mathbf{f}_\mathcal{A}$ назовем такой вектор $\hat{\mathbf{w}}_\mathcal{A}$, который является решением следующей задачи оптимизации:*

$$\hat{\mathbf{w}}_\mathcal{A} = \underset{\mathbf{w}_\mathcal{A} \in \mathbb{R}^k}{\operatorname{argmin}} S(\mathbf{w}_\mathcal{A}|\mathcal{L}). \tag{2.5}$$

Для оценки качества моделей и сравнения их друг с другом введем три критерия качества — сложность, устойчивость и точность.

Определение 2.5. *Сложностью $C = C(\hat{\mathbf{w}})$ модели $\mathbf{f}$ с вектором параметров $\hat{\mathbf{w}} = [w_1, \ldots, w_k]$ назовем мощность множества активных параметров этой модели*

$$C(\mathbf{w}) = \sum_{i=1}^{k} [w_i \neq 0] = |\mathcal{A}|.$$

Чем больше мощность множества активных параметров, тем сложнее модель. Максимально возможная сложность модели равна размерности пространства параметров k.

Определение 2.6. *Устойчивостью $\eta = \eta(\hat{\mathbf{w}})$ модели $\mathbf{f}$ с вектором параметров $\mathbf{w}$ назовем число η, равное числу обусловленности матрицы $\mathbf{A}$ (5.1), т. е.*

$$\eta(\hat{\mathbf{w}}) = \frac{\lambda_{\max}}{\lambda_{\min}},$$

где $\lambda_{\max}$ — максимальное, а $\lambda_{\min}$ — минимальное собственное число матрицы $\mathbf{A}$.

Чем лучше обусловлена матрица $\mathbf{A}$, тем более устойчива модель. У абсолютно устойчивой модели $\lambda_{\min} = \lambda_{\max},\ \eta = 1$.

Определение 2.7. *Под точностью S модели $\mathbf{f}$ с вектором параметров $\hat{\mathbf{w}}$ будем понимать величину функции ошибки (2.3) на контрольной выборке.*

Чем больше значение функции ошибки, тем меньше точность модели.

Введем на множестве допустимых моделей $\mathcal{F}$ отношение доминирования. Будем говорить, что *модель $\mathbf{f}'$ доминирует над моделью $\mathbf{f}$* и обозначать $\mathbf{f}' \succ \mathbf{f}$, если

$$C' \leqslant C, \quad \eta' \leqslant \eta, \quad S' \leqslant S_b,$$

где $C,\ \eta,\ S$ и $C',\ \eta',\ S'$ — сложность, устойчивость и точность моделей $\mathbf{f}$ и $\mathbf{f}'$.

Определение 2.8. *Модель $\mathbf{f} \in \mathcal{F}$ назовем оптимальной по Парето, если не существует $\mathbf{f}' \in \mathcal{F}$ такой, что $\mathbf{f}' \succ \mathbf{f}$.*

Определение 2.9. *Множество оптимальных по Парето моделей назовем Парето-оптимальным фронтом $POF_{\mathfrak{F}}$ множества допустимых моделей $\mathfrak{F}$.*

Задача выбора оптимальной модели состоит в том, чтобы найти Парето-оптимальный фронт $\mathrm{POF}_{\mathfrak{F}}$ множества допустимых моделей $\mathfrak{F}$.

2.2 Путь в k-мерном кубе

В данной задаче будем иметь дело с вектором параметров размерности k. Это означает, что существует 2^k вариантов структуры модели. Из этих 2^k возможных вариантов структуры выбираются оптимальные. Все варианты можно представить в виде вершин $\mathbf{v}$ k-мерного куба $\mathfrak{V}$. И тогда стратегия задает путь $\mathbf{V}$ по его вершинам.

Этот путь заканчивается в некоторой вершине $\hat{\mathbf{v}}$, к которой сходится процедура модификации. Будем искать оптимальные модели в некоторой окрестности вершины $\hat{\mathbf{v}}$. Так как охватить все возможные варианты слишком трудоемко, то в качестве окрестности $\hat{\mathbf{v}}$ будем рассматривать ведущий к $\hat{\mathbf{v}}$ путь по вершинам куба, полученный по описанной выше стратегии.

Пример 1. В этом примере использовалась выборка $\{\mathbf{x}_i, y_i\}$, $i \in \{1, \ldots, 177\}$. Каждый объект выборки описывался 6 признаками $\chi_1, \ldots, \chi_6$ и принадлежал одному из трех классов. Схематично взаимное расположение векторов $\chi_1, \ldots, \chi_6$ изображено на рис. 1. Для классификации такой выборки модифицировалась двухслойная

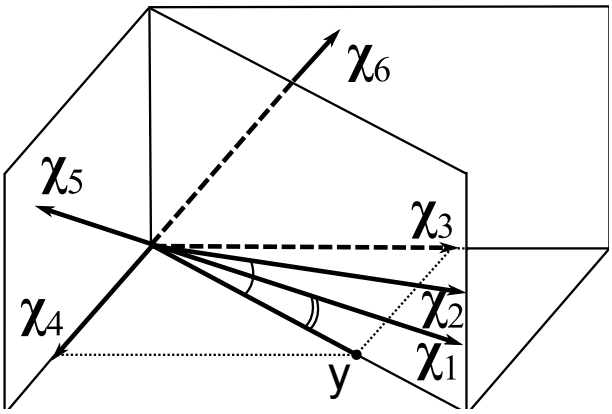


Рис. 1: Данные

нейронная сеть с одним нейроном в скрытом слое. Совокупное число параметров такой сети равно девяти. Нейронная сеть модифицировалась за 11 итераций. На рис. 2 изображен путь по вершинам девятимерного куба. По вертикали отложен номер параметра, по горизонтали — номер итерации. Черная клетка означает, что параметр с индексом j активный, белая клетка — параметр неактивный. Например, на пятой итерации из сети был удален параметр 9, а на одиннадцатой итерации этот параметр был снова добавлен в сеть.

3 Стратегия пошаговой модификации модели

Определение 3.1. *Стратегией пошаговой модификации модели называется процедура последовательного изменения модели, в которой на каждом шаге решается*

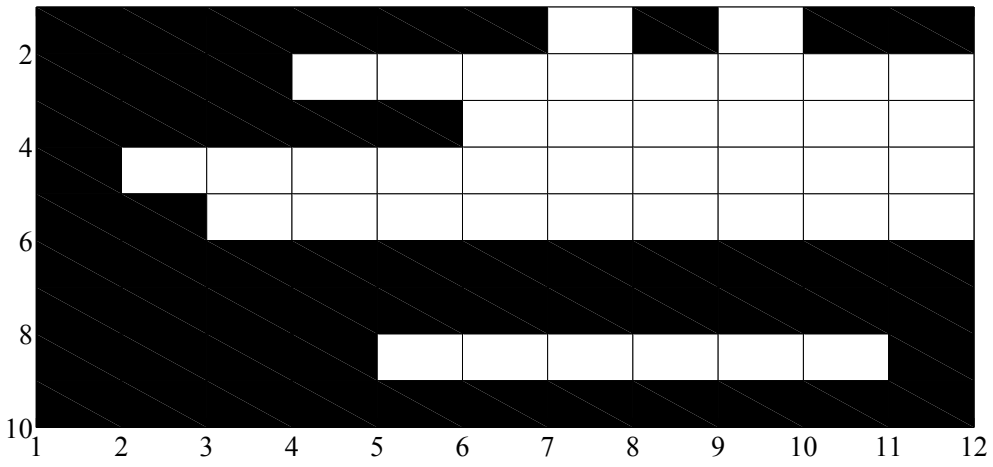


Рис. 2: Путь в кубе

оптимизационная задача вида

$$\hat{j} = \underset{j \in \mathcal{A}}{\operatorname{argopt}} Q(\hat{\mathbf{w}}_{\mathcal{A}}),$$

где Q — один из вышеприведенных критериев качества или их Парето-оптимальный набор.

Стратегия задается следующими математическими объектами:

- набором критериев оптимизации — сложностью, точностью, устойчивостью $\{C, S, \eta\}$,
- набором ограничений на структуру и параметры модели $\mathcal{A} \subseteq \mathcal{J}$, $\mathbf{w} = \hat{\mathbf{w}}_{\mathcal{A}}$ из (2.5),
- критериями останова шагов удаления (3.6) и добавления (3.7),
- критерием останова процедуры выбора модели (3.8).

Действуя согласно стратегии, будем изменять структуру модели, удаляя из нее элементы и добавляя их согласно (3.8).

Для определения индекса параметра $\hat{j}$, который должен быть удален из модели или добавлен в нее, ниже предлагается несколько критериев оптимизации модели.

3.1 Критерий оптимального прореживания

Этот критерий позволяет выяснить индекс параметра, удаление которого приведет к минимизации приращения функции ошибки (2.3). Для функции ошибки ис-

пользуется локальная аппроксимация вблизи некоторого локального минимума вектора параметров $\mathbf{w}_0$:

$$S(\mathbf{w}_0 + \Delta\mathbf{w}) = S(\mathbf{w}_0) + \mathbf{g}^\intercal(\mathbf{w}_0)\Delta\mathbf{w} + \frac{1}{2}\Delta\mathbf{w}^\intercal\mathbf{H}\Delta\mathbf{w} + O(\| \Delta\mathbf{w} \|^3),$$

где $\Delta\mathbf{w}$ — возмущение вектора параметров в данной точке $\mathbf{w}_0$; $\mathbf{g}(\mathbf{w}_0)$ — вектор градиента, вычисленный в точке $\mathbf{w}_0$, $\mathbf{H} = \mathbf{H}(\mathbf{w}_0)$ — матрица вторых производных функции ошибки. Предполагается, что матрица вторых производных $\mathbf{H} = \mathbf{H}(\mathbf{w})$ диагональная, а функция ошибки в окрестности глобального или локального минимума является квадратичной. На основании этих гипотез аппроксимация функции ошибки записывается в следующем виде:

$$\Delta S = S(\mathbf{w}_0 + \Delta\mathbf{w}) - S(\mathbf{w}_0) = \frac{1}{2}\Delta\mathbf{w}^\intercal\mathbf{H}\Delta\mathbf{w}.$$

Пусть w_j — некоторый параметр. Удаление этого параметра (присваивание ему нулевого значения) эквивалентно выполнению условия

$$\mathbf{e}_j^\intercal\Delta\mathbf{w} + w_j = 0,$$

где $\mathbf{e}_j^\intercal$ — вектор, все элементы которого равны нулю, за исключением j-го, который равен единице. Таким образом, получаем задачу условной минимизации

$$\Delta S = \frac{1}{2}\Delta\mathbf{w}^\intercal\mathbf{H}\Delta\mathbf{w} \to \min,\ \mathbf{e}_j^\intercal\Delta\mathbf{w} + w_j = 0.$$

Для решения этой задачи строим лагранжиан

$$L = \frac{1}{2}\Delta\mathbf{w}^\intercal\mathbf{H}\Delta\mathbf{w} - \lambda_i(\mathbf{e}_j^\intercal\Delta\mathbf{w} + w_j).$$

Продифференцировав L по $\Delta\mathbf{w}$, получаем значение лагранжиана L_j для элемента w_j:

$$L_j = \frac{w_j^2}{2[\mathbf{H}^{-1}]_{j,j}},$$

где $\mathbf{H}^{-1}$ — матрица, обратная гессиану $\mathbf{H}$; $[\mathbf{H}^{-1}]_{j,j}$ — j-й диагональный элемент этой матрицы. Значение лагранжиана L_j называется выпуклостью w_j. Выпуклость L_j описывает рост среднеквадратичной ошибки, вызываемый удалением параметра w_j.

Критерию оптимального прореживания отвечает параметр $w_{\hat{j}}$, соответствующий минимальному значению выпуклости:

$$\hat{j} = \operatorname*{argmin}_{j\in\mathcal{A}} L_j. \tag{3.1}$$

3.2 Критерий последовательного прореживания

В качестве второго критерия предлагается простой критерий последовательного удаления параметров w_j — компонент вектора $\mathbf{w}$. Основной идеей этого критерия является принцип локально-оптимального выбора — критерию отвечает параметр w_j, без которого функция ошибки (2.3) оказывается минимальной.

Для нахождения параметра, отвечающего этому критерию, решается задача

$$\hat{j} = \underset{j \in \mathcal{A}}{\operatorname{argmin}} S(\mathbf{w}_{\mathcal{A}} \setminus w_j | \mathcal{T}). \tag{3.2}$$

3.3 Критерий устойчивого прореживания

Помимо вышеописанных критериев предлагается критерий устойчивого прореживания, основанный на модификации метода Белсли [17, 18].

Пусть $\mathbf{W}$ — матрица реализаций оптимального вектора параметров $\hat{\mathbf{w}}$, определенного в (2.5) и рассматриваемого согласно (2.3) как многомерная случайная величина. Пусть эта матрица имеет размерность $r \times k$. Выполним ее сингулярное разложение:

$$\mathbf{W} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathsf{T}}, \tag{3.3}$$

где $\mathbf{U}$ и $\mathbf{V}$ — ортогональные матрицы размера $r \times r$ и $k \times k$, при этом r – количество оценок, а k – размерность вектора параметров $\mathbf{w}$; $\mathbf{\Lambda}$ — матрица, на диагонали которой стоят сингулярные числа матрицы $\mathbf{W}$.

По определению ковариационная матрица вектора параметров $\mathbf{w}$ вычисляется как

$$\mathbf{A}^{-1} = \mathbf{cov}(\mathbf{W}) = \mathsf{E}(\mathbf{W}^{\mathsf{T}}\mathbf{W}) - \mathsf{E}(\mathbf{W})\mathsf{E}(\mathbf{W}^{\mathsf{T}}) = \mathsf{E}(\mathbf{W}^{\mathsf{T}}\mathbf{W}).$$

Последнее равенство выполняется в силу предположения о том, что математическое ожидание вектора параметров равно нулю: $\mathsf{E}(\mathbf{w}) = \mathbf{0}$. По матрице реализаций $\mathbf{W}$ многомерной случайной величины $\mathbf{w}$ ковариационная матрица может быть оценена следующим образом:

$$\mathbf{A}^{-1} = \frac{1}{r}\mathbf{W}\mathbf{W}^{\mathsf{T}}.$$

У ковариационной матрицы есть нулевые строки с индексами из множества $\mathcal{J} \setminus \mathcal{A}$, где $\mathcal{J}$ — множество индексов всех параметров модели, а $\mathcal{A}$ — множество индексов активных параметров. Таким образом, ковариационная матрица является неполноранговой.

Используя сингулярное разложение (3.3) матрицы $\mathbf{W}$ получим выражение для матрицы $\mathbf{A}^{-1}$:

$$\mathbf{A}^{-1} = (\mathbf{W}\mathbf{W}^{\mathsf{T}}) = (\mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\mathsf{T}}\mathbf{V}\mathbf{\Lambda}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}}) = (\mathbf{U}\mathbf{\Lambda}\mathbf{\Lambda}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}}) = \mathbf{U}\mathbf{\Lambda}^2\mathbf{U}^{\mathsf{T}}.$$

Индексом обусловленности η_ζ назовем отношение максимального элемента $\lambda_{\max}$ матрицы $\mathbf{\Lambda}$ к ζ-му по величине элементу λ_ζ этой матрицы:

$$\eta_\zeta = \frac{\lambda_{\max}}{\lambda_\zeta}.$$

Так как ковариационная матрица $\mathbf{A}^{-1}$ неполноранговая, то некоторые значения индексов обусловленности не определены. Чтобы избежать этой проблемы, исключим из рассмотрения параметры с дисперсией, меньшей некоторого порога α, и добавим к каждому элементу, стоящему на диагонали ковариационной матрицы, небольшое число τ.

Оценками дисперсии параметров будут диагональные элементы $\mathbf{A}^{-1}$:

$$\sigma(w_\zeta) = \mathbf{A}^{-1}_{\zeta\zeta}.$$

Долевой коэффицент $q_{\zeta j}$ определим как вклад j-го признака в дисперсию ζ-го элемента вектора параметров $\mathbf{w}$:

$$q_{\zeta j} = \frac{u^2_{\zeta j}\lambda^2_{jj}}{\sigma(w_\zeta)}.$$

Находим индексы обусловленности и долевые коэффициенты для набора активных параметров $\mathcal{A}$. Большие значения индексов обусловленности указывают на зависимость между признаками. Поэтому для нахождения параметра, отвечающего этому критерию прореживания, находим максимальный индекс обусловленности

$$\hat{\zeta} = \operatorname*{argmax}_{\zeta\in\mathcal{A}} \eta_\zeta.$$

Затем находим максимальный долевой коэффициент, соответствующий найденному максимальному индексу обусловленности $\eta_{\hat{\zeta}}$:

$$\hat{j} = \operatorname*{argmax}_{j\in\mathcal{A}} q_{\hat{\zeta}j}. \tag{3.4}$$

Параметр $w_{\hat{j}}$ и есть параметр, отвечающий критерию устойчивого прореживания.

3.4 Критерий последовательного наращивания

Критерий последовательного добавления параметров, как и критерий (3.2), основан на принципе локально-оптимального выбора — критерию отвечает параметр, при добавлении которого в сеть функция ошибки (2.3) минимальна.

Для нахождения параметра, отвечающего этому критерию, решается задача

$$\hat{j} = \underset{j \in \mathcal{J} \setminus \mathcal{A}}{\operatorname{argmin}} S(\mathbf{w}_{\mathcal{A}} \cup w_j | \mathcal{T}). \tag{3.5}$$

3.5 Описание базовой стратегии

Стратегия пошаговой модификации модели состоит из двух этапов — Del и Add. Перед началом процедуры модификации все параметры модели активны.

Этап Del. Ищем параметр с индексом $\hat{j}$, отвечающий одному из критериев прореживания (3.1), (3.2) или (3.3), и удаляем его из множества активных параметров:

$$\mathcal{A} = \mathcal{A} \setminus \hat{j}.$$

Этап Del повторяем до тех пор, пока ошибка $S(\mathbf{w}_{\mathcal{A}}|\mathcal{T})$ не превысит свое минимальное значение на данном этапе более чем на некоторое заданное значение δS_1. Критерием останова шага Del является следующее условие:

$$S(\hat{\mathbf{w}}_{\mathcal{A}}|\mathcal{T}) \geqslant S_{\min} + \delta S_1, \tag{3.6}$$

где $S_{\min}$ — некоторое заданное значение.

Этап Add. В модели ищем параметр $\hat{j}$, отвечающий критерию наращивания (3.4), и добавляем найденный параметр во множество активных параметров:

$$\mathcal{A} = \mathcal{A} \cup \hat{j}.$$

Критерием останова шага Add является выполнение условия

$$S(\hat{\mathbf{w}}_{\mathcal{A}}|\mathcal{T}) \geqslant S_{\min} + \delta S_2, \tag{3.7}$$

где $S_{\min}$ — некоторое заданное значение. На рис. 13 приведен график, демонстрирующий изменение функции ошибки при удалении параметров из модели. Аналогичным образом ведет себя фукнция ошибки при добавлении параметров в модель. Из графика видно, что эта зависимость имеет минимум, а значит модели с бо́льши́м числом параметров не являются наиболее точными. На рис. 14 показано, как согласно критериям останова (3.6) и (3.7) сменяются шаги удаления и добавления. Процедура

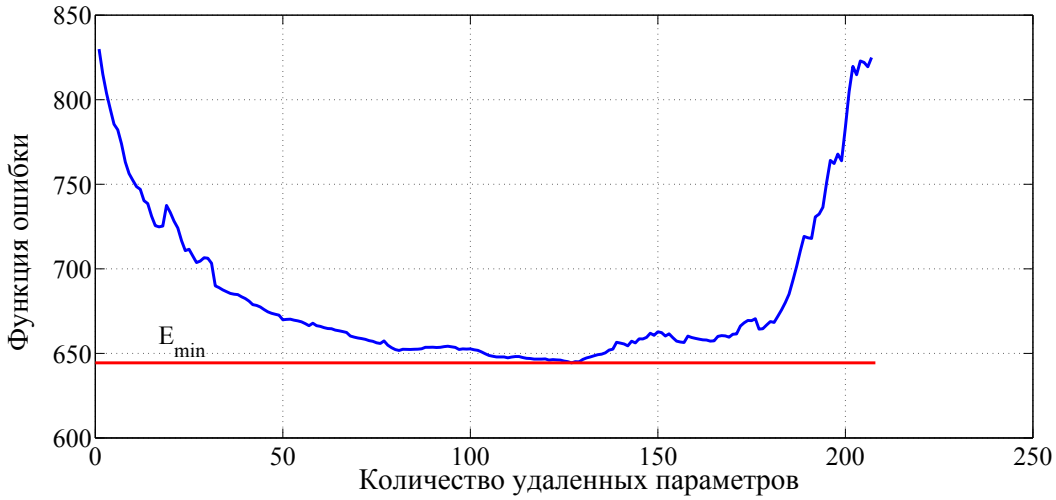


Рис. 3: Изменение функции ошибки при удалении параметров из модели

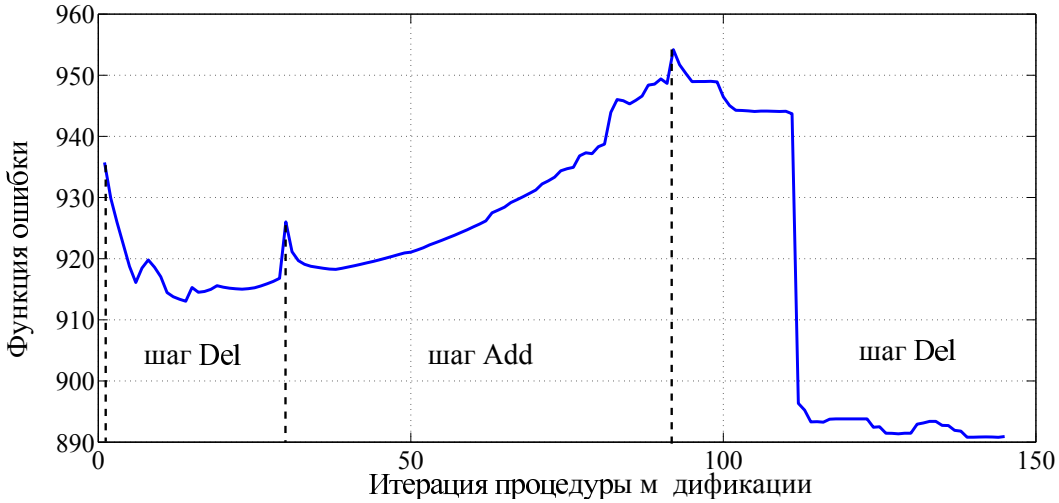


Рис. 4: Смена шагов Del и Add

модификации продолжается до тех пор, пока процесс не стабилизируется. В качестве критерия стабилизации предлагается использовать энтропию изменения структуры модели:

$$H(\mathcal{A}, \mathcal{A}') = -\sum_{j=1}^{k} \rho(a_j, a'_j) \ln(\rho(a_j, a'_j)), \tag{3.8}$$

множества попарных нормированных расстояний Хэмминга между элементами наборов $\mathcal{A} = \{a_1, \ldots, a_k\}$ и $\mathcal{A}' = \{a'_1, \ldots, a'_k\}$, полученных на двух последовательных итерациях алгоритма следующим образом:

$$a_j = \begin{cases} 1, & \text{если } w_j \neq 0, \\ 0, & \text{если } w_j = 0. \end{cases}$$

Процесс считается стабильным, если энтропия $H(\mathcal{A}, \mathcal{A}')$ не превосходит заданного порога.

4 Анализ стратегии пошаговой модификации модели

С целью получить значение критериев качества описанной стратегии был проведен вычислительный эксперимент. Использовались данные акселерометра мобильного телефона. Данные состояли из 5418 векторов признаков, которые были получены в результате обработки соответствующих временных рядов. Было выделено 43 признака и 6 классов физической активности: ходьба, бег, сидение, стояние, подъем и спуск.

Временные ряды записывались акселерометром мобильного телефона, который находился в кармане у человека, выполняющего один из типов физической активности. Для выделения признаков временные ряды разделялись на десятисекундные сегменты. Из этих сегментов извлекались признаки, такие как проекции среднего ускорения на координатные оси, среднеквадратические отклонения от проекций среднего ускорения на каждую из трех координатных осей, время между пиками синусоидального сигнала в миллисекундах и др. С более подробным описанием признаков и процессом их генерации можно ознакомиться в [36].

В вычислительном эксперименте оптимизировалась двухслойная нейронная сеть с пятью нейронами в скрытом слое. Размерность вектора параметров такой модели $k = 245$. Нейронная сеть оптимизировалась по стратегии, описанной в главе 3. Был получен набор из 771 модели. В процедуре модификации использовался каждый из трех критериев прореживания — оптимального, последовательного и устойчивого. Для всех моделей были вычислены значения критериев качества. Был построен Парето-оптимальный фронт трех критериев. На рис. 5 в координатах «устойчивость – сложность» изображены все полученные модели. Синим цветом обозначены модели, которые были получены по стратегии с применением критерия устойчивого прореживания, зеленым цветом — критерия последовательного прореживания, красным цветом — оптимального прореживания. Парето-оптимальные модели обозначены черным крестом. Аналогично на рис. 6 и рис. 7 изображены все полученные модели в координатах «точность – сложность» и «точность – устойчивость» соответственно. Из рис. 5 видно, что самые устойчивые модели получаются при использовании критерия устойчивого прореживания. В таблице приведены значения критериев качества моделей, которые являются точками останова процедуры модификации для каждого их трех критериев прореживания.

Таблица 1: Сложность, точность, устойчивость моделей

Стратегия	Сложность	Точность	Устойчивость
Оптимальное прореживавние	50	877	$1,2 \cdot 10^6$
Последовательное прореживавние	36	870	$2 \cdot 10^6$
Устойчивое прореживание	50	866	$6 \cdot 10^5$

На рис. 8 приведена интерпретация полученных результатов. В красной области

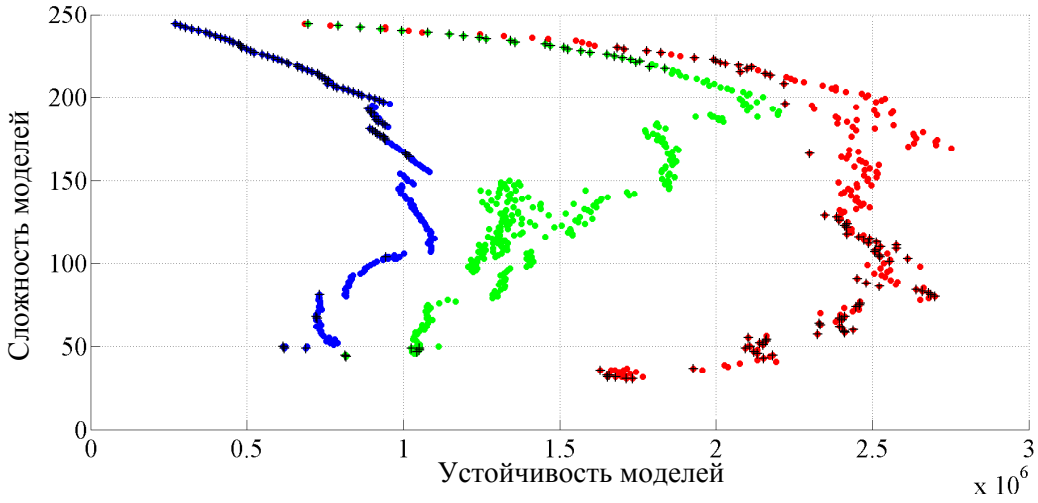


Рис. 5: Множество моделей в координатах устойчивость – сложность

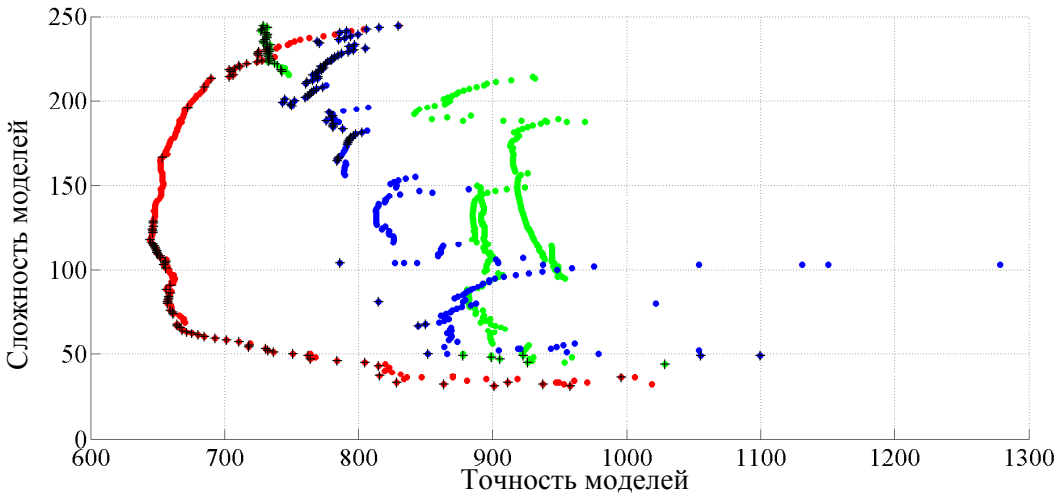


Рис. 6: Множество моделей в координатах точность – сложность

графика Парето-оптимальные модели не интересны для рассмотрения, так как в этой области имеет место недообучение — модели излишне сложны. Парето-оптимальные модели с незначительной сложностью находятся в зеленой области графика.

Также была визуализирована процедура пошаговой модификации модели как путь в k-мерном кубе. На рис. 9–11, так же как и в примере 1, по вертикали отложен номер параметра, по горизонтали — номер итерации. Черная клетка означает, что параметр активный, белая клетка — параметр неактивный. На рис. 8–10 указана последовательность, в которой параметры удалялись из модели и добавлялись в нее. Из рис. 10, 11 видно, что стратегия с критериями оптимального и последовательного прореживания, которые выбирают для удаления параметр, минимизирующий функ-

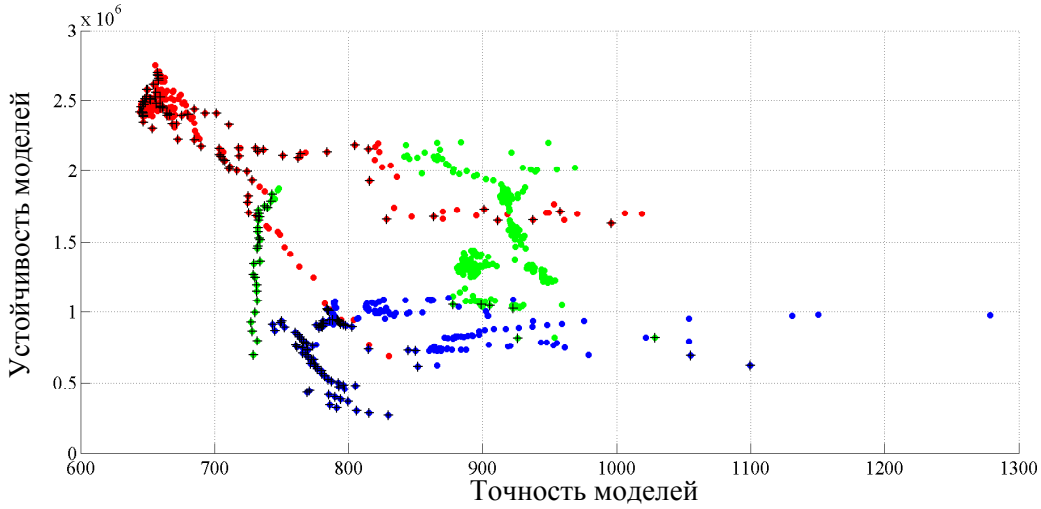


Рис. 7: Множество моделей в координатах точность – устойчивость

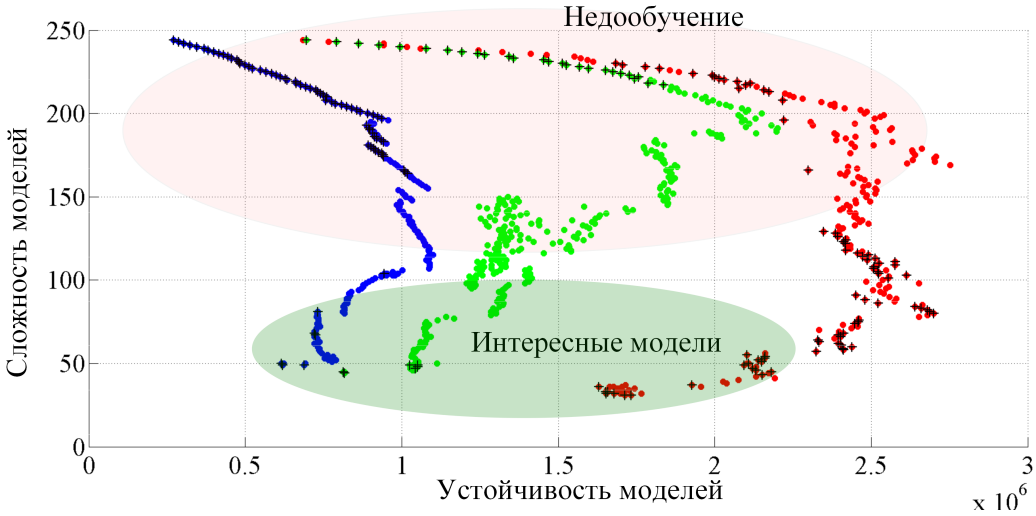


Рис. 8: Интерпретация результатов

цию ошибки, оставляет в моделях параметры с номерами с 216 по 245. Это связано с тем, что параметры с такими номерами относятся ко второму слою нейронной сети, а удаление большого числа параметров второго слоя приводит к росту функции ошибки.

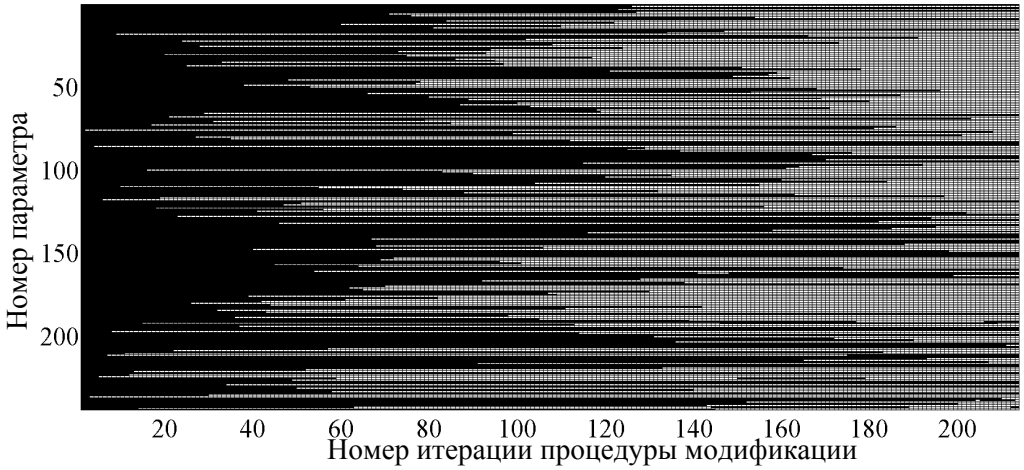


Рис. 9: Путь в кубе, устойчивое прореживание

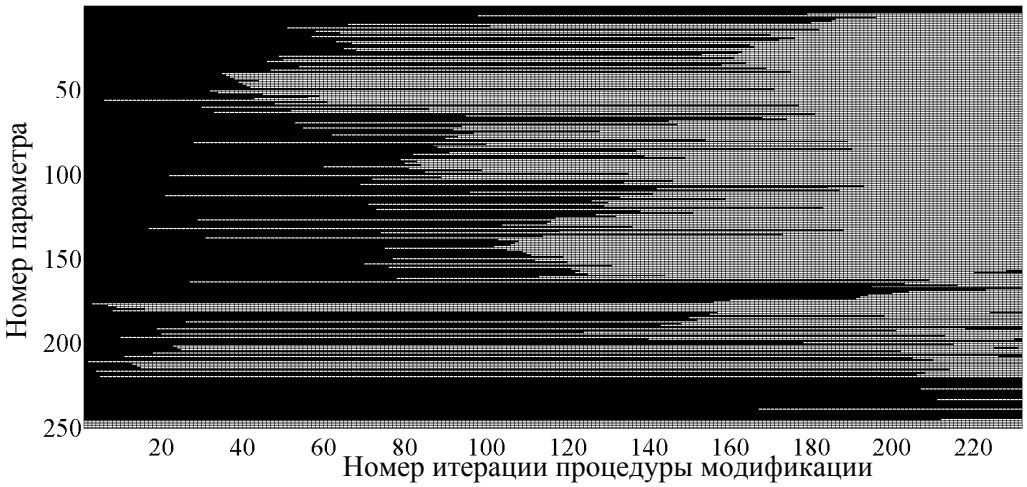


Рис. 10: Путь в кубе, последовательное прореживание

5 Постановка задачи построения суперпозиции нейронных сетей

Работа посвящена решению задачи классификации временных рядов с использованием нейронных сетей глубокого обучения. Предлагается использовать многоуровневую суперпозицию моделей, относящихся к следующим классам нейронных сетей: двухслойные нейронные сети, машины Больцмана и автокодировщики. Нижние уровни суперпозиции выделяют из зашумленных данных высокой размерности информативные признаки, а верхний уровень по этим признакам решает задачу классификации. Предложенная модель была протестирована на двух выборках времен-

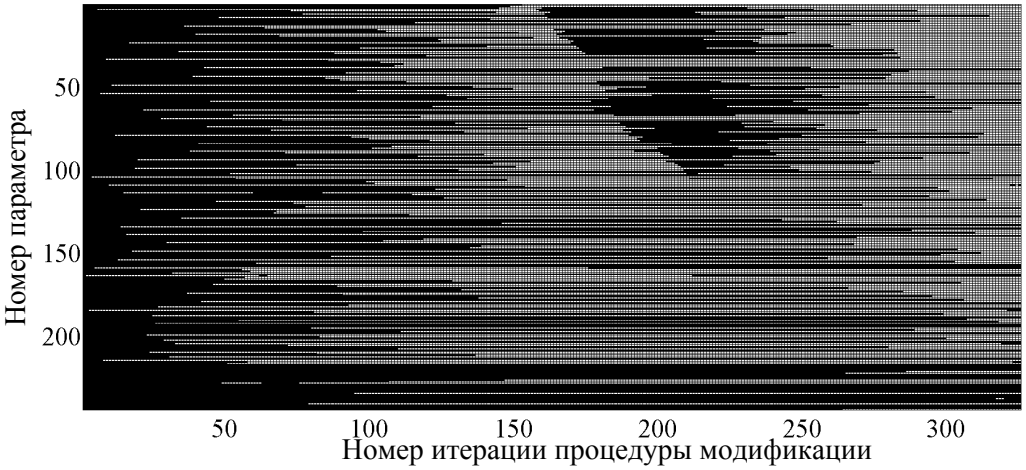


Рис. 11: Путь в кубе, оптимальное прореживание

ных рядов физической активности человека. Результаты классификации, полученные предлагаемой моделью в ходе вычислительного эксперимента, сравнивались с результатами, которые были получены на этих же данных в работах зарубежных авторов. Исследование показало возможность применения нейронных сетей глубокого обучения к решению прикладных задач классификации физической активности человека.

В данной работе рассматривается задача классификации временных рядов. Под временным рядом понимается упорядоченный набор измерений некоторой величины, в котором каждое измерение соответствует определенному моменту времени. Временные ряды обладают уникальными свойствами [20], которые усложняют работу с ними. Примерами таких свойств служат высокая размерность и зашумленность данных. Существуют методы снижения размерности и фильтрации шумов [2-4]. Однако при снижении размерности пространства данных и выделении новых признаков снижается точность описания объектов. Ввиду этого возникает задача построения нового признакового пространства меньшей размерности, в котором признаки наиболее полно описывали бы исходные временные ряды.

Существует два основных способа построения признакового пространства – это экспертное и автоматическое выделение признаков. Первый способ заключается в экспертном назначении базовых функций и требует индивидуального подхода к каждой отдельной задаче, т. к. одни и те же базовые функции не могут достаточно точно описывать данные разной природы. Второй способ является более универсальным, и некоторые методы, успешно применяемые для предобработки временных рядов,

описаны в [5-7]. В данной работе предлагается использовать нейронные сети глубокого обучения для выделения информативных признаков [7-9] и классификации временных рядов.

Нейронные сети глубокого обучения применяются для решения задач распознавания изображений [29] и речи [30], однако есть работы, показывающие возможность их применения к предобработке и классификации временных рядов [31, 32]. Предлагается использовать многоуровневую суперпозицию [26] ограниченных машин Больцмана [33, 34], автокодировщиков [34] и двухслойных нейронных сетей для машинного извлечения признаков и классификации временных рядов. Все уровни суперпозиции, кроме последнего, обучаются по принципу «обучение без учителя» и участвуют в построении признакового пространства. Последний уровень суперпозиции обучается «с учителем» и решает задачу классификации по признакам, выделенным на нижних уровнях суперпозиции. Такая конструкция позволяет снизит размерность пространства признаков, информативно описывающих исследуемое явление, а затем решить задачу классификации, основываясь на небольшом числе выделенных признаков. В данной работе поставлен вычислительный эксперимент на двух выборках – временных рядах акселерометра и временных рядах акселерометра и гироскопа мобильного телефона. Результаты классификации, полученные с помощью предложенной модели, сравнивались с результатами, полученными на тех же данных в других работах [35, 36]. Исследование показало возможность применения нейронных сетей глубокого обучения к решению прикладной задачи классификации физической актисновсти человека.

Дана выборка $\mathfrak{D} = \{(\mathbf{x}_i, t_i), i = 1, \ldots, N\}$, состоящая из N пар объект–ответ. Объектами $\mathbf{x}_i$ являются сегменты временного ряда — $\mathbf{x}_i \in \mathbb{R}^n$. Каждый объект принадлежит одному из M классов, метки классов $t_i \in \{1, \ldots, M\}$. Выборка $\mathfrak{D}$ разделена на две подвыборки — обучающую $\mathfrak{L}$ и контрольную $\mathfrak{T}$.

Моделью классификации назовем суперпозицию функций

$$\mathbf{f}(\mathbf{x}, \mathbf{w}) = \boldsymbol{\mu}_1(\boldsymbol{\mu}_2(\ldots \boldsymbol{\mu}_K(\mathbf{x}))) : \mathbb{R}^n \to [0, 1]^M, \tag{5.1}$$

где $\boldsymbol{\mu}_k,\ k \in \{1, \ldots, K\}$, – модели из класса нейронных сетей с соответствующими векторами параметров $\mathbf{w}_k,\ k \in \{1, \ldots, K\}$. Под вектором параметров модели $\mathbf{f}$ будем понимать вектор $\mathbf{w} = [\mathbf{w}_1, \ldots, \mathbf{w}_K]^\intercal$.

Компоненты вектора $\mathbf{f}(\mathbf{x}, \mathbf{w})$ – это вероятности отнести объект $\mathbf{x}_i$ к соответству-

ющему классу:

$$\mathbf{f}(\mathbf{x}, \mathbf{w}) = \begin{bmatrix} p(y_1 = 1|\mathbf{x}; \mathbf{w}) \\ p(y_2 = 2|\mathbf{x}; \mathbf{w}) \\ \vdots \\ p(y_M = M|\mathbf{x}; \mathbf{w}) \end{bmatrix}. \tag{5.2}$$

Функцию ошибки S на некоторой подвыборке $\mathfrak{K}$ исходной выборки $\mathfrak{D}$ определим следующим образом:

$$S(\mathbf{w}|\mathfrak{K}) = -\frac{1}{|\mathfrak{K}|} \sum_{i=1}^{|\mathfrak{K}|} \sum_{\xi=1}^{M} [y_\xi = \xi] \log p(y_\xi = \xi|\mathbf{x}_i, \mathbf{w}). \tag{5.3}$$

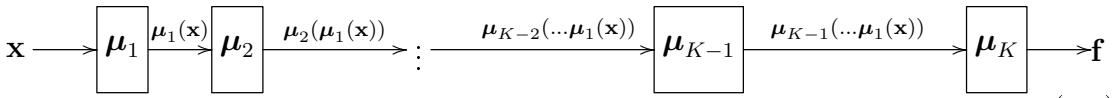
Для решения задачи классификации выборки $\mathfrak{D}$ с помощью модели нужно оптимизировать ее параметры $\mathbf{w}$. Для этого требуется решить задачу минимизации функции ошибки на обучающей выборке:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\operatorname{argmin}}\, S(\mathbf{w}|\mathfrak{L}). \tag{5.4}$$

Для дополнительной оценки качества классификации будем вычислять значения функционала AUC (area under curve) на контрольной выборке для каждого класса по принципу один против всех и визуализировать полученные результаты с помощью ROC-кривых.

6 Описание алгоритма построения суперпозиции нейронных сетей

Согласно (5.1) предлагаемая модель представляет из себя K-уровневую суперпозицию нейронных сетей (6.1). В данной работе будем рассматривать нейронные сети следующих типов: ограниченная машина Больцмана, автокодировщик и двухслойная нейронная сеть. Далее будет описан каждый из вышеперечисленных типов нейронных сетей.



(6.1)

6.1 Ограниченная машина Больцмана

Ограниченная машина Больцмана – это двухслойная нейронная сеть. Структура ограниченной машины Больцмана представляет из себя двудольный граф. Нейроны

первого слоя называются видимыми и соответствуют значениям признаков объекта. Вектор состояний нейронов первого слоя – это вектор $\mathbf{v} = \{v_l \in \mathbb{R}, l \in \text{vis}\}$, а $\text{vis} = \{1, \dots L\}$ – множество индексов видимых нейронов. Нейроны второго слоя называются скрытыми и участвуют в выделении признаков. В данной работе нейроны скрытого слоя могут находиться в одном из двух состояний — 0 или 1. Вектор состояний – это вектор $\mathbf{h} = \{h_j \in [0,1], j \in \text{hid}\}$, $\text{hid} = \{1, \dots, J\}$ – множество индексов скрытых нейронов. Каждая компонента h_j вектора состояний – это вероятность того, что j-й нейрон скрытого слоя находится в состоянии 1:

$$h_j = p(j\text{-й нейрон в состоянии 1}). \tag{6.2}$$

Таким образом, ограниченная машина Больцмана — это вероятностная модель. Вероятность пары векторов состояний нейронов $(\mathbf{v}, \mathbf{h})$ определяется следующим образом:

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h})), \tag{6.3}$$

где $E(\mathbf{v}, \mathbf{h})$ – это значение энергии пары $(\mathbf{v}, \mathbf{h})$, а Z – нормировочная величина, которая определяется следующим образом:

$$Z = \sum_{\mathbf{v},\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})).$$

Выражение для энергии пары $E(\mathbf{v}, \mathbf{h})$ зависит от типа объектов, которые требуется моделировать ограниченной машинной Больцмана. В данной работе используется энергия для моделирования бинарных данных:

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{l \in \text{vis}} b_l^v v_l - \sum_{j \in \text{hid}} b_j^h h_j - \sum_{l,j} v_l h_j w_{lj} \tag{6.4}$$

или, как вариант, энергия для моделирования вещественных данных:

$$E(\mathbf{v}, \mathbf{h}) = \sum_{l \in \text{vis}} \frac{(v_l - b_l^v)^2}{2\sigma_l^2} - \sum_{j \in \text{hid}} b_j^h h_j - \sum_{l,j} \frac{v_l}{\sigma_l} h_j w_{lj}, \tag{6.5}$$

где σ_l – стандартное нормальное отклонение шума l-го признака, параметры $b_l^v, b_j^h, l \in \text{vis}, j \in \text{hid}$, – смещения нейронов видимого и скрытого слоев и $\mathbf{W} = [w_{lj}], l \in \text{vis}, j \in \text{hid}$ – матрица весовых коэффициентов между нейронами видимого и скрытого слоев.

Вероятность входного вектора состояний $\mathbf{v}$, или вероятность того, что входной вектор $\mathbf{v}$ описывается моделью (6.3), выражается как сумма по всем скрытым состояниям:

$$p(\mathbf{v}) = \sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})).$$

Приведем выражения для условных вероятностей [34], которые понадобятся далее для оптимизации параметров:

$$p(\mathbf{h}|\mathbf{v}) = \frac{p(\mathbf{v},\mathbf{h})}{p(\mathbf{v})} = \frac{\exp(-E(\mathbf{v},\mathbf{h}))}{\sum_{\mathbf{h}}\exp(-E(\mathbf{v},\mathbf{h}))} = \prod_{j\in\text{hid}} p(h_j|\mathbf{v}),$$

$$p(\mathbf{v}|\mathbf{h}) = \frac{p(\mathbf{v},\mathbf{h})}{p(\mathbf{h})} = \frac{\exp(-E(\mathbf{v},\mathbf{h}))}{\sum_{\mathbf{v}}\exp(-E(\mathbf{v},\mathbf{h}))} = \prod_{l\in\text{vis}} p(v_l|\mathbf{h}).$$

Оптимизация параметров ограниченной машины Больцмана заключается в нахождении таких значений параметров $\boldsymbol{\Theta} = \{w_{lj}, b_l^v, b_j^h, l \in \text{vis}, j \in \text{hid}\}$, при которых величина вероятности элементов обучающей выборки имеет наибольшее значение:

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, p(\mathcal{L};\boldsymbol{\Theta}) = \prod_{\mathbf{x}\in\mathcal{L}} \frac{1}{Z}\sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})}.$$

Выражение для энергии (6.5) используется, если соответствующая машина Больцмана является первым уровнем суперпозиции (5.1). В остальных случаях используется выражение (6.4).

6.2 Алгоритм оптимизации параметров ограниченной машины Больцмана

Приведем краткое описание алгоритма оптимизации параметров ограниченной машины Больцмана с одношаговым сэмплированием Гиббса. С подробным описанием и обоснованием можно ознакомиться в [34].

Ниже приведен псевдокод одного шага алгоритма. Один цикл оптимизации ограниченной машины Больцмана состоит в повторении этого шага для всех объектов обучающей выборки. Предлагается выполнять некоторое заданное количество циклов.

Исходные параметры: $\mathbf{x}$ – входной вектор признаков, $\mathbf{W}$, $\mathbf{b}^v$, $\mathbf{b}^h$ – начальные значения параметров, см. (**??**).

Результат: $\mathbf{W}$, $\mathbf{b}^v$, $\mathbf{b}^h$ – значения параметров после одного шага оптимизации.

Инициализация

$\mathbf{v}_1 = \mathbf{x}$;

для каждого $j \in$ hid **выполнять**

  вычислить вероятность $p(h_{1j} = 1|\mathbf{v}_1)$;

  выбрать значение h_{1j} из множества $\{0, 1\}$ в зависимости от величины $p(h_{1j}|\mathbf{v}_1)$;

конец цикла

для каждого $k \in$ vis **выполнять**

  вычислить вероятность $p(v_{2k} = 1|\mathbf{h}_1)$;

  выбрать значение v_{2k} из множества $\{0, 1\}$ в зависимости от величины вероятности $p(v_{2i}|\mathbf{h}_1)$;

конец цикла

для каждого $j \in$ hid **выполнять**

  вычислить $P(h_{2j} = 1|\mathbf{v}_2)$;

конец цикла

$\mathbf{W} \leftarrow \mathbf{W} + \eta(\mathbf{h}_1\mathbf{v}_1^\intercal - p(\mathbf{h}_{2*} = 1|\mathbf{v}_1)\mathbf{v}_2^\intercal)$;

$\mathbf{b}^v \leftarrow \mathbf{a} + \eta(\mathbf{v}_1 - \mathbf{v}_2)$;

$\mathbf{b}^h \leftarrow \mathbf{b} + \eta(\mathbf{h}_1 - p(\mathbf{h}_{2*} = 1|\mathbf{v}_2))$.

6.3 Автокодировщик

Автокодировщик представляет собой следующую суперпозицию блоков:

$$\boldsymbol{\mu} = \boldsymbol{\varphi}(\mathbf{g}(\mathbf{x})),$$

где

$$\mathbf{g}(\mathbf{x}) = \boldsymbol{\sigma}(\mathbf{W}_g\mathbf{x} + \mathbf{b}_g) - \text{кодируюший блок, или encoder,} \tag{6.6}$$

$$\boldsymbol{\varphi}(\mathbf{g}(\mathbf{x})) = \boldsymbol{\sigma}(\mathbf{W}_h\mathbf{g}(\mathbf{x}) + \mathbf{b}_h) - \text{декодирующий блок, или decoder,} \tag{6.7}$$

а $\mathbf{W}_g$, $\mathbf{W}_\varphi$, $\mathbf{b}_g$, $\mathbf{b}_\varphi$ – параметры автокодировщика, $\boldsymbol{\sigma}(\mathbf{t}) = \dfrac{1}{1 + \exp(-\mathbf{t})}$ – сигмоидная функция.

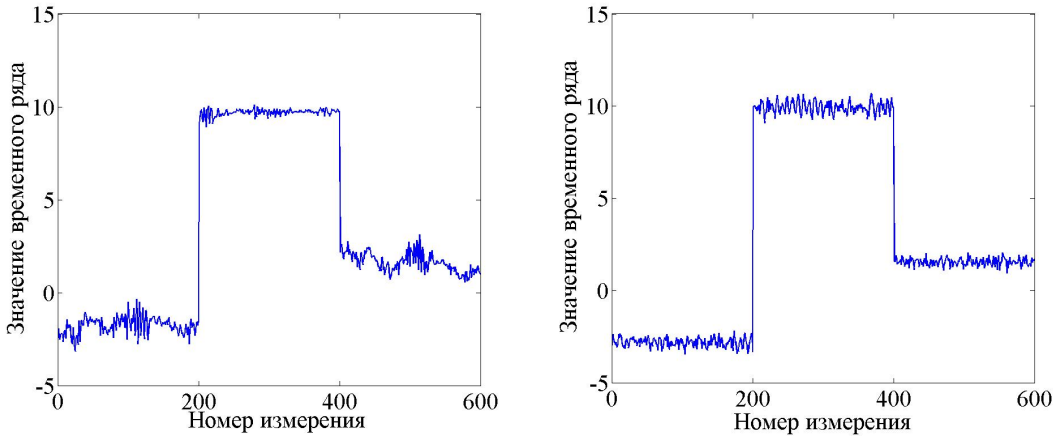


Рис. 12: Прообраз (слева) и восстановленный прообраз (справа) сегмента временного ряда

Под образом вектора $\mathbf{x}$ будем понимать вектор $\mathbf{g}(\mathbf{x}) = \sigma(\mathbf{W}_g\mathbf{x} + \mathbf{b}_g)$.

Будем предполагать, что в данной модели матрицы $\mathbf{W}_g$ и $\mathbf{W}_\varphi$ ортогональны:

$$\mathbf{W}_\varphi = \mathbf{W}_g^\intercal.$$

Оптимизация параметров автокодировщика $\mathbf{\Theta} = (\mathbf{W}_g, \mathbf{W}_h, \mathbf{b}_g, \mathbf{b}_h)$ проводится так, чтобы по прообразу $\mathbf{g}(\mathbf{x})$ можно было восстановить образ $\mathbf{x}$ с помощь преобразования (6.7) или, другими словами, чтобы выходной вектор $\mathbf{f}$ был как можно больше похож на входной вектор $\mathbf{x}$ для всех элементов обучающей выборки. Мерой сходства в данной работе выступает следующая функция:

$$S(\mathbf{\Theta}, \mathbf{x}) = ||\mathbf{f}(\mathbf{x}|\mathbf{\Theta}) - \mathbf{x}||_2^2.$$

Таким образом,

$$\hat{\mathbf{\Theta}} = \underset{\mathbf{\Theta}}{\operatorname{argmin}} \frac{1}{2|\mathcal{L}|} \sum_{\mathbf{x}\in\mathcal{L}} S(\mathbf{\Theta}, \mathbf{x}). \tag{6.8}$$

Для оптимизации параметров автокодировщика нужно выбрать начальное приближение для параметров каждого блока в отдельности [33], а затем настроить параметры всей модели как целого методом обратного распространения ошибки [26].

На рис. 1 изображены прообраз и восстановленный прообраз сегмента временного ряда с помощью автокодировщика.

6.4 Двухслойная нейронная сеть

Двухслойная нейронная сеть – это отображение вида

$$\mathbf{a}(\mathbf{x}) = \mathbf{W}_2^\intercal \mathbf{tanh}\left(\mathbf{W}_1^\intercal \mathbf{x}\right), \tag{6.9}$$

$$\boldsymbol{\mu}(\mathbf{x}) = \frac{\mathbf{exp}(\mathbf{a}(\mathbf{x}))}{\sum_{j=1}^{n} \mathbf{exp}(a_j(\mathbf{x}))}.$$

Вектор $\boldsymbol{\mu}$ интерпретируется как вектор вероятностей: μ_ξ есть вероятность того, что вектор $\mathbf{x}$ принадлежит классу с номером ξ:

$$\boldsymbol{\mu}(\mathbf{x}) = \{\mu_\xi\}, \quad 0 \leqslant \mu_\xi \leqslant 1, \quad \sum \mu_\xi = 1, \quad \xi \in \{1, \ldots, M\}.$$

Под вектором параметров двухслойной нейронной сети будем понимать $\mathbf{w} = \mathbf{vec}(\mathbf{W}_1^\intercal | \mathbf{W}_2^\intercal)$, где $\mathbf{W}_1, \mathbf{W}_2$ — матрицы весов первого и второго слоя нейронной сети (6.9). Вектор меток классов $\mathbf{y} = [y_1, \ldots, y_\xi, \ldots, y_M]^\intercal$ определим следующим образом:

$$y_\xi = \begin{cases} 1, & \text{если } \xi = \underset{\xi \in \{1,\ldots,M\}}{\operatorname{argmax}} (f_\xi), \\ 0 & \text{иначе.} \end{cases}$$

В качестве функции ошибки выберем функцию

$$S(\mathbf{w}|\mathfrak{K}) = -\sum_{\mathbf{x} \in \mathfrak{K}} \sum_{\xi=1}^{M} [y_t = 1] \ln\big(f_\xi(\mathbf{x}, \mathbf{w})\big), \tag{6.10}$$

максимизирующую логарифм правдоподобия мультиномиально распределенной случайной величины $\mathbf{y}$ и заданную на подвыборке $\mathfrak{K}$ исходной выборки $\mathfrak{D}$, t – истинная метка класса.

Оптимизация параметров двухслойной нейронной сети заключается в том, чтобы найти вектор параметров $\mathbf{w}$, минимизирующий функцию ошибки S по обучающей выборке. В данной работе оптимизация параметров проводится методом обратного распространения ошибки.

7 Анализ суперпозиции нейронных сетей глубокого обучения

В вычислительном эксперименте использовались два набора временных рядов с датчиков мобильного устройства – WISDM [36] и HAR [35]. Цель вычислительного эксперимента заключалась в повышении точности классификации для различных типов суперпозиции моделей и сравнении их со значениями точности из работ [36, 35]. В вычислительном эксперименте также получены значения AUC для каждого из классов и построены соответствующие ROC-кривые.

7.1 Программное обеспечение

Для построения модели как суперпозиции блоков, описанных в предыдущем разделе, было реализовано программное обеспечение на языке MatLab. Для этого были использованы инструментарии [18-21]. На рис. 13 и 14 изображены схемы в стандарте IDEF0, описывающие структуру проекта.

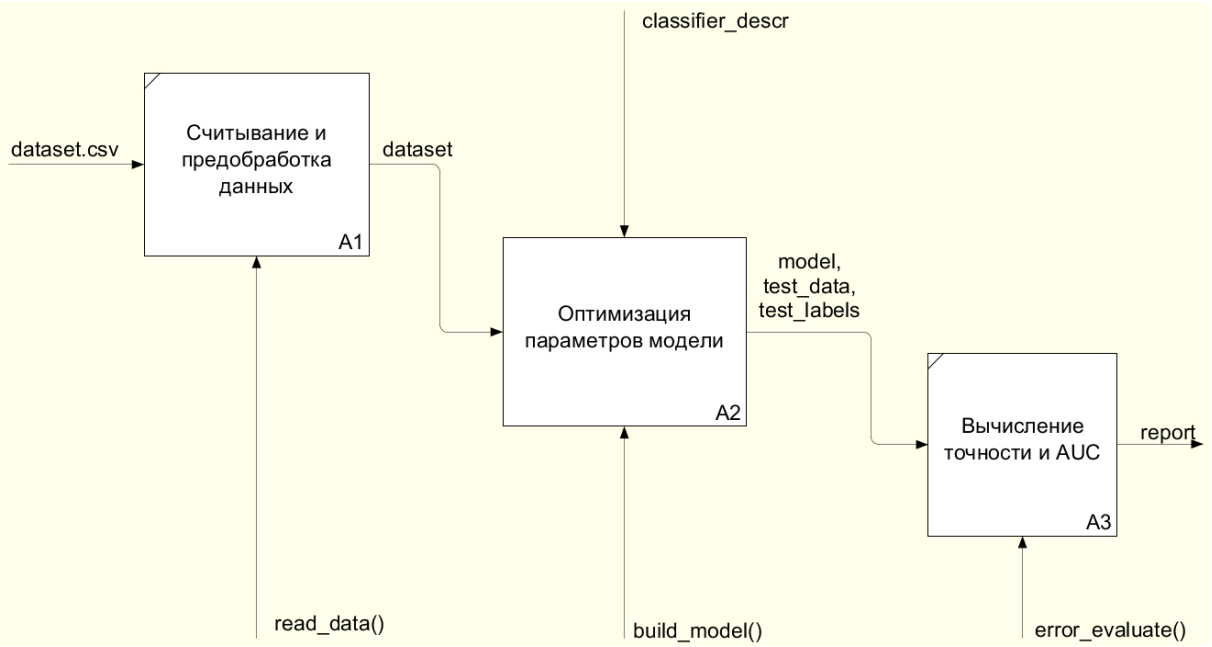


Рис. 13: Общая структура проекта

7.2 Эксперимент на наборе данных WISDM

Первый набор состоял из сегментов временных рядов акселерометра мобильного телефона, каждый из которых описывал один из четырех типов физической активности человека – ходьбу, бег, стояние и сидение. Сегмент представлял из себя 10 секундный отрезок исходного временного ряда и состоял из 600 измерений – по 200 измерений проекции ускорения на каждую из координатных осей. С более подробным описанием данных можно ознакомиться в работе [36].

Выборка состояла из 4219 объектов и была неоднородной по числу элементов в разных классах. Для того чтобы избежать такого эффекта, как игнорирование моделью малочисленных классов, в обучающую выборку добавлялись повторяющие объекты таким образом, чтобы сбалансировать число представителей каждого класса. Разделение на обучающую и контрольную выборку проводилось случайным об-

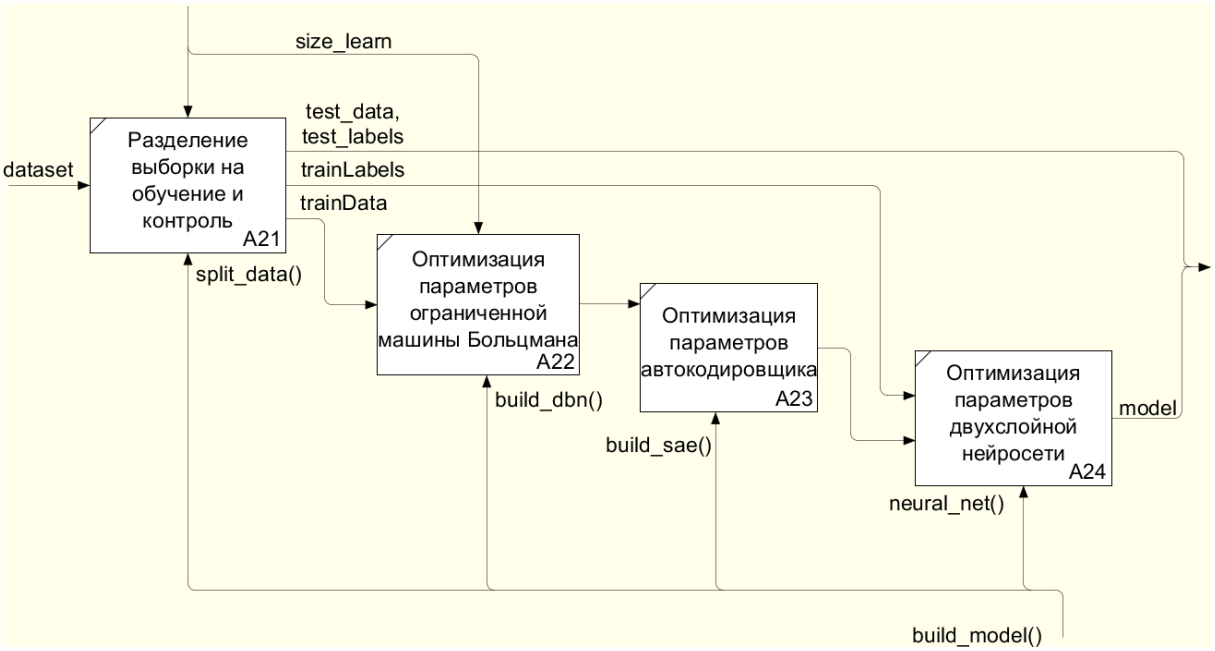


Рис. 14: Структура блока A2 Оптимизация параметров модели

разом в соотношении 3:1. В эксперименте использовались следующие суперпозиции: трехслойная суперпозиция, состоящая из машины Больцмана, автокодировщика и двухслойной нейросети с 300, 200 и 100 нейронами в каждом слое соответственно; четырехслойная суперпозиция, состоящая из машины Больцмана, еще одной машины Больцмана, автокодировщика и двухслойной нейросети с 400, 300, 200 и 100 нейронами в каждом слое соответственно; а также пятислойная суперпозиция, состоящая из двух машин Больцмана, двух автокодировщиков и двухслойной нейросети с 500, 400, 300, 200 и 100 нейронами в каждом слое соответственно. В табл. 2 приведены результаты точности классификации для описанных выше моделей, а также результаты из работы [36]. В табл. 3 приведены значения функционала AUC, а на рис. 4 – соответствующие ROC-кривые.

Таблица 2: Сравнительные результаты

Класс	3 слоя	4 слоя	5 слоев	Kwapisz et. al.
Бег	98%	95%	97%	98%
Ходьба	95%	94%	96%	92%
Сидение	100%	100%	100%	95%
Стояние	89%	82%	84%	92%

Таблица 3: значения функционала AUC

Класс	Бег	Ходьба	Сидение	Стояние
3 слоя	0,985	0,964	0,999	0,902
4 слоя	0,983	0,964	0,990	0,960
5 слоев	0,981	0,939	1,000	0,822

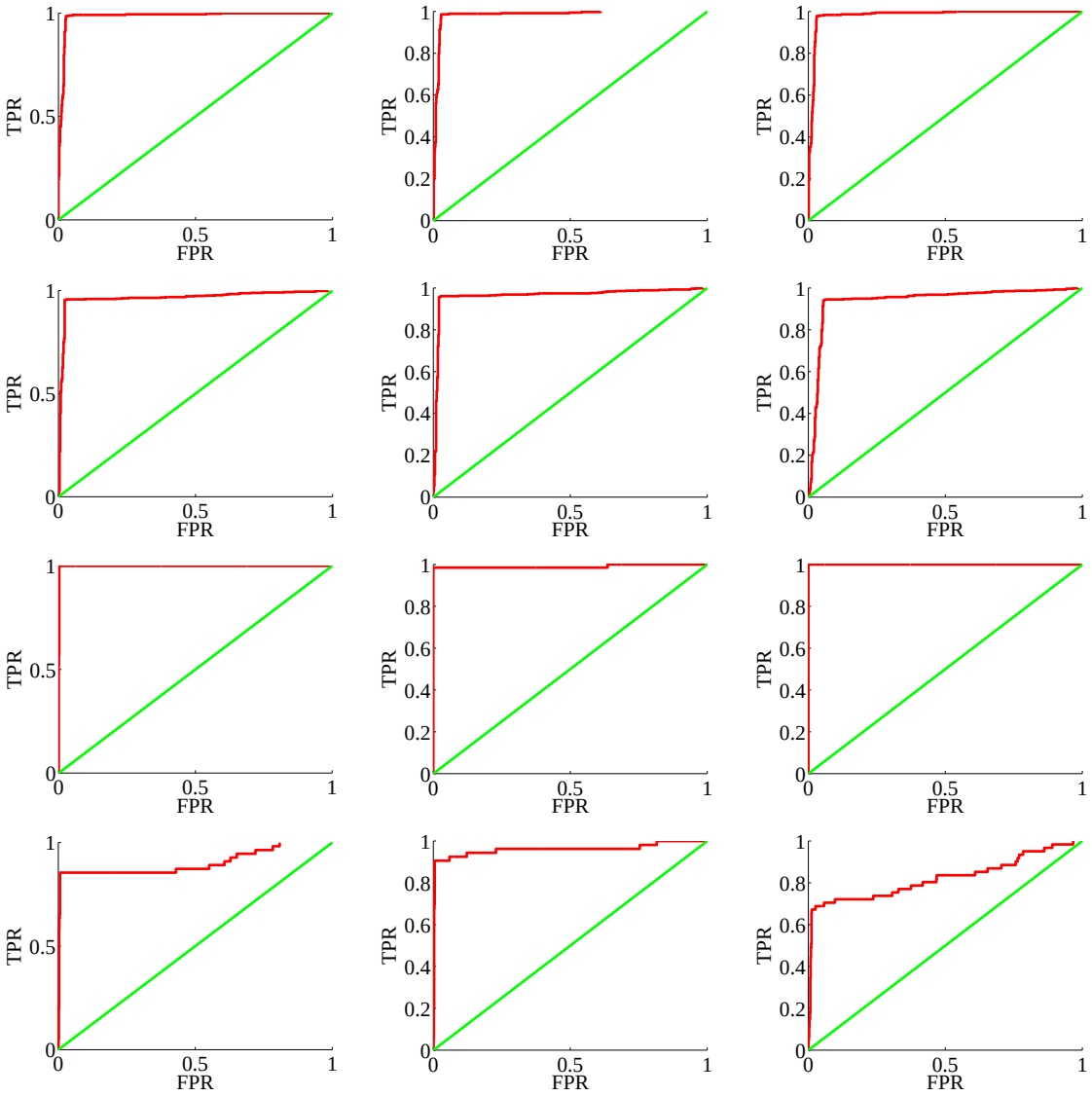


Рис. 15: ROC-кривые для каждого класса (по строчкам): суперпозиция из 3 слоев (слева), из 4 слоев (в центре), из 5 слоев (справа)

7.3 Эксперимент на наборе данных HAR

Второй набор состоял из векторов признаков, полученных предварительной обработкой сегментов временных рядов с акселерометра и гироскопа мобильного телефона Samsung Galaxy S II. Временные ряды сегментировались на 2,56-секундные отрезки, из которых затем получали вектор из 561 признака. Более подробное описание данных приведено в [35]. В этой части вычислительного эксперимента, как и в предыдущей, использовались следующие суперпозиции: трехслойная суперпозиция, состоящая из машины Больцмана, автокодировщика и двухслойной нейросети с 300, 200 и 100 нейронами в каждом слое соответственно, четырехслойная суперпозиция, состоящая из машины Больцмана, машины Больцмана, автокодировщика и двухслойной нейросети с 400, 300, 200 и 100 нейронами в каждом слое соответственно, а также пятислойная суперпозиция, состоящая из двух машин Больцмана, двух автокодировщиков и двухслойной нейросети с 500, 400, 300, 200 и 100 нейронами в каждом слое соответственно. В табл. 4 приведены результаты точности классификации для описанных выше моделей, а также результаты из работы [35]. В табл. 5 приведены значения функционала AUC, а на рис. 5 – соответствующие ROC-кривые.

Таблица 4: Сравнительные результаты

Класс	3 слоя	4 слоя	5 слоев	Anguita et. al.
Ходьба	96%	98%	96%	97%
Подъём	94%	93%	80%	87%
Спуск	98%	92%	96%	72%
Сидение	91%	65%	84%	95%
Стояние	75%	80%	71%	97%
Лежание	99,7%	98%	92%	100%

8 Заключение

В работе была предложена стратегия пошаговой модификации моделей классификации согласно трем критериям качества — сложности, точности и устойчивости. В рамках стратегии были предложены критерии добавления и удаления парамет-

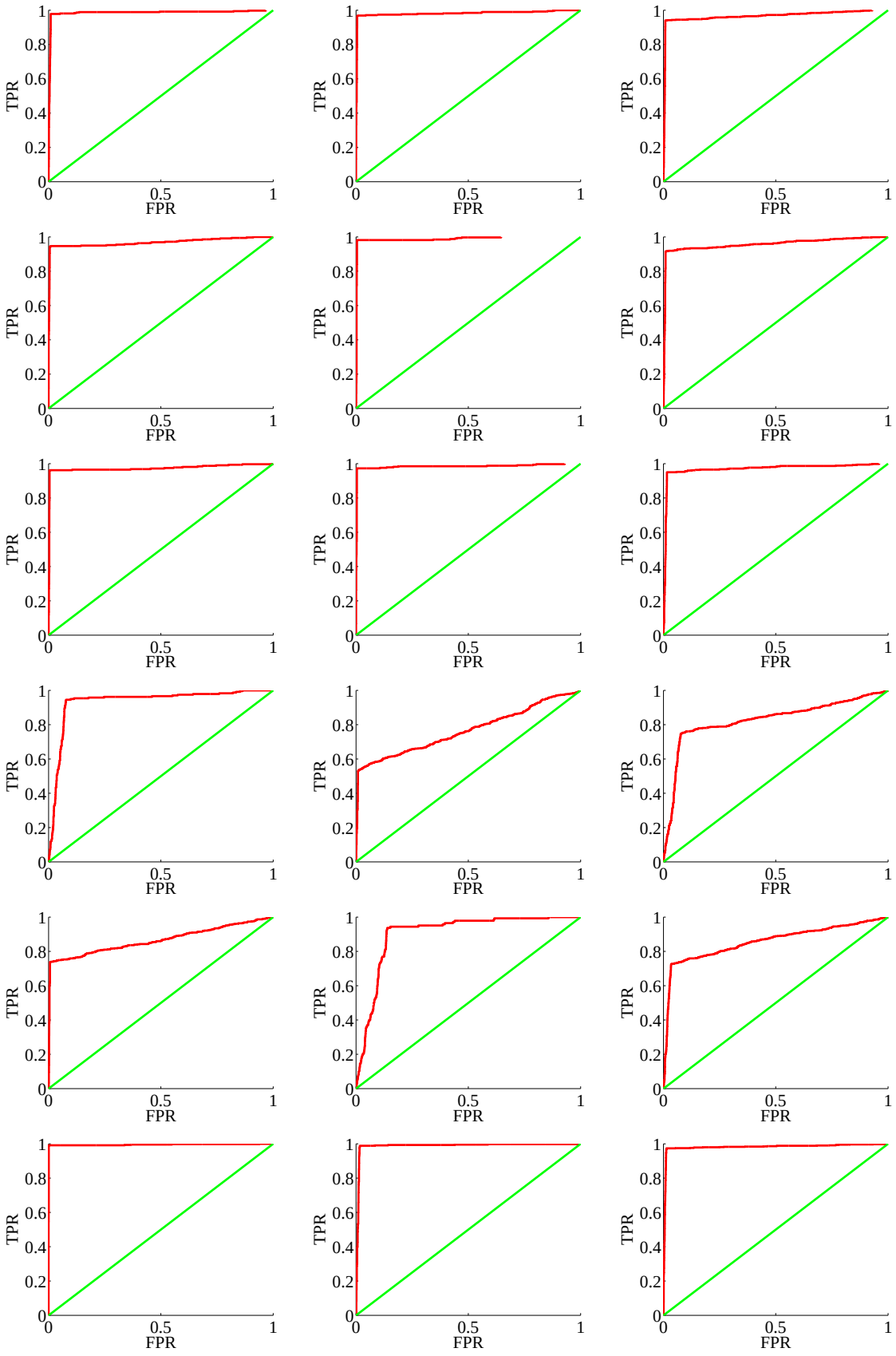


Рис. 16: ROC-кривые для каждого класса (по строчкам): суперпозиция из 3 слоев (слева), из 4 слоев (в центре), из 5 слоев (справа)

Таблица 5: Значения функционала AUC

Класс	Бег	Ходьба	Сидение	Стояние	Сидение	Стояние
3 слоя	0,986	0,969	0,974	0,934	0,866	0,995
4 слоя	0,982	0,989	0,984	0,767	0,903	0,987
5 слоев	0,9672	0,958	0,969	0,828	0,858	0,981

ров в модель, критерии останова шагов добавления и удаления, а также критерий останова процедуры модификации. Процедура пошаговой модификации модели была рассмотрена и визуализирована как путь в многомерном кубе. Был проведен вычислительный эксперимент, в ходе которого был получен набор моделей и найден Парето-оптимальный фронт критериев качества этого набора. Вычислительный эксперимент показал, что наилучшие по рассматриваемым критериям качества модели получаются при использовании критерия устойчивого прореживания. Это связано с тем, что критерий устойчивого прореживания позволяет получать более устойчивые модели, удаляя коррелирующие параметры и тем самым повышая устойчивость и обобщающую способность модели классификации. Программная реализация стратегии пошаговой модификации нейронной сети в среде разработки MatLab находится в свободном доступе по адресу [19].

В данной работе решалась задача классификации временных рядов. В качестве модели классификации была предложена суперпозиция нейронных сетей глубокого обучения. В вычислительном эксперименте рассматривалась задача многоклассовой классификации физической активности по измерениям с датчиков мобильного телефона. Для выполнения вычислительного эксперимента было реализовано программное обеспечение на языке MatLab, агрегирующее в себе несколько инструментариев. Точность классификации предложенной моделью вычислялась на наборах данных из [35, 36] и сравнивалась с точностью, полученной в этих работах. Полученные результаты оказались сравнимы с результатами из соответствующих работ, что говорит о возможности применения суперпозиции нейронных сетей глубокого обучения к решению задачи классификации временных рядов.

9 Литература

Список литературы

[1] *Визильтер Ю. В., Горбацевич В. С., Каратеев С. Л., Костромов Н. А.* Обучение алгоритмов выделения кожи на цветных изображениях лиц // Информатика и ее применения, 2012. Т. 6. Вып. 1. С. 109–113.

[2] *Хапланов А. Ю.* Асимптотическая нормальность оценки параметров многомерной логистической регрессии // Информатика и ее применения, 2013. Т. 7. Вып. 2. С. 69–74.

[3] *Токмакова А. А., Стрижов В. В.* Оценивание гиперпараметров линейных и регрессионных моделей при отборе шумовых и коррелирующих признаков // Информатика и ее применения, 2012. Т. 6. Вып. 4. С. 66–75.

[4] *Myung I. J.* The Importance of Complexity in Model Selection // Journal of Mathematical Psychology, 2000. Vol. 44. No. 1. P. 190–204.

[5] *MacLeod C., Maxwell M.* Incremental evolution in ANNs: Neural nets which grow // Artificial Intelligence Review, 2001. Vol. 16. No. 3. P. 201–224.

[6] *Karnin E. D.* A simple procedure for pruning back-propagation trained neural networks // IEEE Transactions on Neural Networks, 1990. Vol. 1. No. 2. P. 239–242.

[7] *LeCun Y., Denker L. S., Solla S. A.* Optimal Brain Damage // Advances in neural information processing systems, 1990. Vol. 2. No. 2. P. 598–605.

[8] *Hassibi B., Stork D. G., Woff G. J.* Optimal brain surgeon and general network pruning // Proceedings of 1993 IEEE International Conference on Neural Networks, 1993. Vol. 1. P. 293–299.

[9] *Yang S., Chen Y.* An evolutionary constructive and pruning algorithm for artificial neural networks and its prediction applications // Neurocomputing, 2012. Vol. 86. P. 140–149.

[10] *Pu X., Pengfei Sun P.* A New Hybrid Pruning Neural Network Algorithm Based on Sensitivity Analysis for Stock Market Forcast // Journal of Information and Computational Science, 2013. Vol. 3. P. 883–892.

[11] *Hong-Gui H., Qi-li C., Jun-Fei Q.* An efficient self-organizing RBF neural network for water quality prediction // Neural Networks, 2011. Vol. 24. No. 7. P. 717–725.

[12] *Knerr S., Personnaz L., Dreyfus G.* Single-layer learning revisited: a stepwise procedure for building and training a neural network // Neurocomputing: Algorithms, Architectures and Applications, 1990. Vol. 68. No. 1. P. 41–50.

[13] *Strijov V., Krymova E., Weber S. V.* Evidence optimization for consequently generated models // Mathematical and Computer Modelling, 2010. Vol. 57. No. 1–2. P. 50–56.

[14] *Леонтьева Л. Н.* Последовательный выбор признаков при восстановлении регрессии // Машинное обучение и анализ данных, 2012. Т. 1. № 3. С. 335–346.

[15] *Зайцев А. А., Токмакова А. А.* Оценка гиперпараметров линейных регрессионных моделей методом максимального правдоподобия при отборе шумовых и коррелирующих признаков // Машинное обучение и анализ данных, 2012. Т. 1. № 3. С. 347–353.

[16] *Kwapisz J. R., Weiss G. M., Moore S.* Activity recognition using cell phone accelerometers // SIGKDD Explorations, 2010. Vol. 12. No 2. P. 74–82.

[17] *Belsley D. A., Kuh E., Welsch R. E.* Regression diagnostics: Identifying influential data and sources of collinearity. – New York: John Wiley and Sons, 2005.

[18] *Сандуляну Л. Н., Стрижов В. В.* Выбор признаков в авторегрессионных задачах прогнозирования // Информационные технологии, 2012. Т. 7. С. 11–15.

[19] *Попова М. С.* Реализация стратегии пошаговой модификации нейронной сети // Algorithms of Machine Learning, 2014. http://sourceforge.net/p/mlalgorithms/code/HEAD/tree/Group174/Popova2014OptimalModelSelec

[20] *Längkvist M., Karlsson L., Loutfi A.* A Review of Unsupervised Feature Learning and Deep Learning for Time-Series Modeling // Pattern Recognition Letters, 2014. Vol. 42. No. 6. P. 11–24.

[21] *Nanopoulos A., Alcock R., Manolopoulos Y.* Feature-based Classification of Time-series Data // International Journal of Computer Research, 2001. Vol. 10. P. 49–61.

[22] *Keogh E., Pazzani M.* A simple dimensionality reduction technique for fast similarity search in large time series databases // Proceedings of Pacific-Asia Conf. on Knowledge Discovery and Data Mining. – Kyoto, Japan: Springer, 2000. P. 122–133.

[23] *Weston J., Mukherjee S., Chapelle O., Pontil M., Poggio T., Vapnik V.* Feature selection for SVMs //Advances in neural information processing systems. – Denver, USA: MIT Press, 2000. P. 668—674.

[24] *Estévez P. A., Tesmer M., Perez C. A., Zurada J. M.* Normalized Mutual Information Feature Selection // IEEE Transactions on Neural Networks, 2009. Vol. 20. No. 2. P. 189–201.

[25] *Mörchen F.* Time series feature extraction for data mining using DWT and DFT // OAI-PMH server at citeseerx.ist.psu.edu, 2003. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.87.2037.

[26] *Hinton G. E., Salakhutdinov R. R.* Reducing the dimensionality of data with neural networks // Science, 2006. Vol. 313. No. 5786. P. 504–507.

[27] *Stuhlsatz A., Lippel J., Zielke T.* Feature Extraction With Deep Neural Networks by a Generalized Discriminant Analysis // IEEE Transactions on Neural Networks and Learning Sysems, 2012. Vol. 23. No. 4. P. 596–608.

[28] *Ren Y., Wu Y.* Convolutional deep belief networks for feature extraction of EEG signal // Proceedings of International Joint Conference on Neural Networks (IJCNN 2014). – Beijing, China: IEEE, 2014. P. 2850–2853.

[29] *Xu Y., Mo T., Feng Q., Zhong P., Lai M., Chang C.* Deep learning of feature representation with multiple instance learning for medical image analysis // Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP 2014). – Florence, Italy: IEEE, 2014. P. 1626–1630.

[30] *Hinton G. E., Li Deng, Dong Yu, DahlG. E., Mohamed A., Jaitly N., Senior A., Vanhoucke V., Nguyen P., Sainath T. N., Kingsbury B.* Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups // IEEE Signal Process. Mag., 2012. Vol. 29. No. 6. P. 82–97.

[31] *Hüsken M., Stagge P.* Recurrent Neural Networks for Time Series Classification // Neurocomputing, 2003. Vol. 50. P. 223–235.

[32] *Wulsin D., Gupta J., Mani R., Blanco J., Litt B.* Modeling electroencephalography waveforms with semi-supervised deep belief nets: faster classification and anomaly measurement // Journal of Neural Engineering, 2011. Vol. 8. P. 1741–2552.

[33] *Hinton G. E.* A Practical Guide to Training Restricted Boltzmann Machines // Neural Networks: Tricks of the Trade. 2nd ed. – Springer, 2012. P. 599–619.

[34] *Bengio Y.* Learning Deep Architectures for AI // Foundations and Trends in Machine Learning, 2009. Vol. 2. No. 1. P. 1–127.

[35] *Anguita D., Ghio A., Oneto L., Parra X., Luis Reyes-Ortiz J.* Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine // Ambient Assisted Living and Home Care: Proceedings of the 4th International Workshop (IWAAL 2012). – Springer, 2012. Vol. 7657. P. 216–223.

[36] *Kwapisz J. R., Weiss G. M., Moore S.* Activity recognition using cell phone accelerometers // SIGKDD Explorations, 2010. Vol. 12. No 2. P. 74–82.

[37] *Nabney I.* NETLAB: algorithms for pattern recognitions. – Springer-Verlag, 2002.

[38] Deep Learning Toolbox https://github.com/zellyn/deeplearning-class-2011/tree/master/ufldl/library

[39] Deep Neural Network http://www.mathworks.com/matlabcentral/fileexchange/42853-deep-neural-network

[40] Deep Learning Toolbox https://github.com/rasmusbergpalm/DeepLearnToolbox