

Министерство образования Республики Беларусь
Учреждение образования «Витебский государственный
университет имени П.М. Машерова»
Кафедра прикладного и системного программирования

Е.А. Корчевская

ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

Методические рекомендации

*Витебск
ВГУ имени П.М. Машерова
2022*

УДК 004.8(075.8)
ББК 16.6я73
К70

Печатается по решению научно-методического совета учреждения образования «Витебский государственный университет имени П.М. Машерова». Протокол № 1 от 05.10.2022.

Автор: доцент кафедры прикладного и системного программирования ВГУ имени П.М. Машерова, кандидат физико-математических наук, доцент **Е.А. Корчевская**

Рецензент:
заведующий кафедрой «Математика и информационные технологии» УО «ВГТУ», кандидат физико-математических наук, доцент *Т.В. Никонова*

Корчевская, Е.А.

К70 Интеллектуальный анализ данных : методические рекомендации / Е.А. Корчевская. – Витебск : ВГУ имени П.М. Машерова, 2022. – 44 с.

В методических рекомендациях рассматриваются теоретические основы интеллектуального анализа данных, включающие характеристику основных этапов анализа и начальные сведения об используемом математическом аппарате.

Приведена краткая теоретическая информация, предложен список заданий для самостоятельного выполнения. Данное издание предназначается для студентов специальностей «Управление информационными ресурсами» (дисциплина «Интеллектуальный анализ данных»), «Прикладная информатика» (дисциплина «Методы искусственного интеллекта»).

УДК 004.8(075.8)
ББК 16.6я73

© Корчевская Е.А., 2022
© ВГУ имени П.М. Машерова, 2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1. Интеллектуальный анализ данных	5
2. Нейросетевое моделирование	10
2.1 Общая характеристика нейронных сетей	10
2.2 Обучение нейронных сетей	18
3. Анализ временных рядов и прогнозирование	26
3.1 Постановка задачи	26
3.2 Использование метода скользящей средней в прогнозировании	32
3.3 Методы наименьших квадратов	35
ЛИТЕРАТУРА	43

ВВЕДЕНИЕ

Методические рекомендации посвящены интеллектуальному анализу данных, который в настоящее время широко используется в различных сферах. Благодаря интеллектуальному анализу данных можно достичь двух результатов: описать имеющиеся данные или сделать прогнозы на будущее. Материал разбит на 3 раздела.

В первом разделе «Интеллектуальный анализ данных» приведена классификация задач DataMining, а также их практическое применение.

Во втором разделе «Нейросетевое моделирование» описана общая характеристика нейронных сетей, методы обучения нейронных сетей, а также предложены задания для самостоятельного выполнения.

В третьем разделе «Анализ временных рядов и прогнозирование» рассмотрены методы анализа временных рядов и способы моделирования тенденции. Предложены задания для самостоятельного выполнения.

Материал соответствует отдельным темам рабочих программ курсов: «Интеллектуальный анализ данных» (специальность «Управление информационными ресурсами»), «Методы искусственного интеллекта» (специальность «Прикладная информатика»).

1. Интеллектуальный анализ данных

DataMining – исследование и обнаружение «машиной» (алгоритмами, средствами искусственного интеллекта) в сырых данных скрытых знаний, которые ранее не были известны, нетривиальны, практически полезны, доступны для интерпретации человеком.

Классификация задач *DataMining*

Методы ДМ помогают решить многие задачи, с которыми сталкивается аналитик. Из них основными являются: классификация, регрессия, поиск ассоциативных правил и кластеризация.

Задача классификации сводится к определению класса объекта по его характеристикам. Необходимо заметить, что в этой задаче множество классов, к которым может быть отнесен объект, заранее известно.

Задача регрессии, подобно задаче классификации, позволяет определить по известным характеристикам объекта значение некоторого его параметра. В отличие от задачи классификации значением параметра является не конечное множество классов, а множество действительных чисел.

При поиске ассоциативных правил целью является нахождение частых зависимостей (или ассоциаций) между объектами или событиями. Найденные зависимости представляются в виде правил и могут быть использованы как для лучшего понимания природы анализируемых данных, так и для предсказания появления событий.

Задача кластеризации заключается в поиске независимых групп (кластеров) и их характеристик во всем множестве анализируемых данных. Решение этой задачи помогает лучше понять данные. Кроме того, группировка однородных объектов позволяет сократить их число, а следовательно, и облегчить анализ.

Перечисленные задачи по назначению делятся на описательные и предсказательные.

Описательные (descriptive) задачи уделяют внимание улучшению понимания анализируемых данных. Ключевой момент в таких моделях – легкость и прозрачность результатов для восприятия человеком. Возможно, обнаруженные закономерности будут специфической чертой именно конкретных исследуемых данных и больше нигде не встретятся, но это все равно может быть полезно и потому должно быть известно. К такому виду задач относятся кластеризация и поиск ассоциативных правил.

Решение предсказательных (predictive) задач разбивается на два этапа. На первом этапе на основании набора данных с известными результатами строится модель. На втором этапе она используется для предсказания результатов на основании новых наборов данных. При этом, естественно, требуется, чтобы построенные модели работали максимально точно. К данному виду задач относят задачи классификации и регрессии. Сюда можно отнести и задачу поиска ассоциативных правил, если результаты ее решения могут быть использованы для предсказания появления некоторых событий.

Задача кластеризации состоит в разделении исследуемого множества объектов на группы «похожих» объектов, называемых кластерами. Часто решение задачи разбиения множества элементов на кластеры называют кластерным анализом.

Кластеризация может применяться практически в любой области, где необходимо исследование экспериментальных или статистических данных. Рассмотрим пример из области маркетинга, в котором данная задача называется сегментацией.

Концептуально сегментирование основано на предпосылке, что все потребители – разные. У них разные потребности, разные требования к товару, они ведут себя по-разному: в процессе выбора товара, в процессе приобретения товара, в процессе использования товара, в процессе формирования реакции на товар. В связи с этим необходимо по-разному подходить к работе с потребителями: предлагать им различные по своим характеристикам товары, по-разному продвигать и продавать товары. Для того чтобы определить, чем отличаются потребители друг от друга и как эти отличия отражаются на требованиях к товару, и производится сегментирование потребителей.

Практическое применение DataMining.

Интернет -технологии

В системах электронного бизнеса, где особую важность имеют вопросы привлечения и удержания клиентов, технологии DataMining часто применяются для построения рекомендательных систем интернет-магазинов и для решения проблемы персонализации посетителей сайтов. Рекомендации товаров и услуг, построенные на основе закономерностей в покупках клиентов, обладают огромной убеждающей силой.

Торговля

Для успешного продвижения товаров всегда важно знать, что и как продается, а также, кто является потребителем. Исчерпывающий ответ на первый вопрос дают такие средства DataMining, как анализ рыночных корзин и сиквенциальный анализ. Зная связи между покупками и временные закономерности, можно оптимальным образом регулировать предложение. С другой стороны, маркетинг имеет возможность непосредственно управлять спросом, но для этого необходимо знать как можно больше о потребителях –целевой аудитории маркетинга. DataMining позволяет решать задачи выделения групп потребителей со схожими стереотипами поведения, т.е. сегментировать рынок. Для этого можно применять такие технологии DataMining, как кластеризацию и классификацию

Телекоммуникации

Телекоммуникационный бизнес является одной из наиболее динамически развивающихся областей современной экономики. Возможно, поэтому традиционные проблемы, с которыми сталкивается в своей деятельности любая компания, здесь ощущаются особо остро.

Промышленное производство

Промышленное производство создает идеальные условия для применения технологий DataMining. Причина – в самой природе технологического процесса, который должен быть воспроизводимым и контролируемым. Все отклонения в течение процесса, влияющие на качество выходного результата, также находятся в заранее известных пределах. Таким образом, создается статистическая стабильность, первостепенную важность которой отмечают в работах по классификации. Естественно, что в таких условиях использование DataMining способно дать лучшие результаты, чем, к примеру, при прогнозировании ухода клиентов телекоммуникационных компаний.

Медицина

В медицинских и биологических исследованиях, равно как и в практической медицине, спектр решаемых задач настолько широк, что возможно использование любых методологий DataMining. Примером может служить построение диагностической системы или исследование эффективности хирургического вмешательства.

Известно много экспертных систем для постановки медицинских диагнозов. Они построены главным образом на основе правил, описывающих сочетания различных симптомов отдельных заболеваний. С помощью таких правил узнают не только, чем болен пациент, но и как нужно его лечить. Правила помогают выбирать средства медикаментозного воздействия, определять показания / противопоказания, ориентироваться в лечебных процедурах, создавать условия наиболее эффективного лечения, предсказывать исходы назначенного курса лечения и т.п. Технологии DataMining позволяют обнаруживать в медицинских данных шаблоны, составляющие основу указанных правил.

Банковское дело

Классическим примером использования DataMining на практике является решение проблемы о возможной некредитоспособности клиентов банка. Этот вопрос, тревожащий любого сотрудника кредитного отдела банка, можно разрешить и интуитивно.

Страховой бизнес

В страховании, так же как в банковском деле и маркетинге, возникает задача обработки больших объемов информации для определения типичных групп (профилей) клиентов. Эта информация используется для того, чтобы предлагать определенные услуги страхования с наименьшим для компании риском и, возможно, с пользой для клиента.

Другие области применения

DataMining может применяться практически везде, где возникает задача автоматического анализа данных. В качестве примера приведем такие популярные направления, как анализ и последующая фильтрация спама, а также разработка так называемых виртуальных собеседников.

Базовые методы DataMining

К базовым методам DataMining принято относить прежде всего алгоритмы, основанные на переборе. Простой перебор всех исследуемых объектов требует $O(2^N)$ операций, где N – количество объектов. Следовательно, с увеличением количества данных объем вычислений растет экспоненциально, что при большом объеме делает решение любой задачи таким методом практически невозможным.

Основные этапы анализа

Для обнаружения знаний в данных недостаточно просто применить методы DataMining, хотя, безусловно, этот этап является основным в процессе интеллектуального анализа. Весь процесс состоит из нескольких этапов. Рассмотрим основные из них, чтобы продемонстрировать, что без специальной подготовки аналитика методы DataMining сами по себе не решают существующих проблем.

Итак, весь процесс можно разбить на следующие этапы

- понимание и формулировка задачи анализа;
- подготовка данных для автоматизированного анализа (препроцессинг);
- применение методов DataMining и построение моделей;
- проверка построенных моделей;
- интерпретация моделей человеком.

На первом этапе выполняется осмысление поставленной задачи и уточнение целей, которые должны быть достигнуты методами DataMining. Важно правильно сформулировать цели и выбрать необходимые для их достижения методы, т.к. от этого зависит дальнейшая эффективность всего процесса.

Второй этап состоит в приведении данных к форме, пригодной для применения конкретных методов DataMining.

Третий этап – это собственно применение методов DataMining. Сценарии этого применения могут быть самыми различными и включать сложную комбинацию разных методов, особенно если используемые методы позволяют проанализировать данные с разных точек зрения.

Следующий этап – проверка построенных моделей. Очень простой и часто используемый способ заключается в том, что все имеющиеся данные, которые необходимо анализировать, разбиваются на две группы. Как правило, одна из них большего размера, другая – меньшего.

Последний этап – интерпретация полученных моделей человеком в целях их использования для принятия решений, добавление получившихся правил и зависимостей в базы знаний и т.д. Этот этап часто подразумевает использование методов, находящихся на стыке технологии DataMining и технологии экспертных систем.

Кластеризация

Кластеризация отличается от классификации тем, что для проведения анализа не требуется иметь выделенную целевую переменную, с этой точки

зрения она относится к классу *unsupervised learning*. Эта задача решается на начальных этапах исследования, когда о данных мало что известно. Ее решение помогает лучше понять данные, и с этой точки зрения задача кластеризации является описательной задачей (*descriptive*).

Для этапа кластеризации характерно отсутствие каких-либо различий как между переменными, так и между записями. Напротив, ищутся группы наиболее близких, похожих записей. Методы автоматического разбиения на кластеры редко используются сами по себе, просто для получения групп схожих объектов. Анализ только начинается с разбиения на кластеры. После определения кластеров используются другие методы *DataMining*, для того чтобы попытаться установить, а что означает такое разбиение на кластеры, чем оно вызвано.

Большое достоинство кластерного анализа в том, что он позволяет производить разбиение объектов не по одному параметру, а по целому набору признаков. Кроме того, кластерный анализ, в отличие от большинства математико-статистических методов, не накладывает никаких ограничений на вид рассматриваемых объектов и позволяет рассматривать множество исходных данных практически произвольной природы.

Формально задача кластеризации описывается следующим образом.

Дано множество объектов данных I , каждый из которых представлен набором атрибутов. Требуется построить множество кластеров C и отображение F множества I на множество C , т. е. $F: I \rightarrow C$. Отображение F задает модель данных, являющуюся решением задачи. Качество решения задачи определяется количеством верно классифицированных объектов данных.

Множество I определим следующим образом:

$$I = \{i_1, i_2, \dots, i_j, \dots, i_n\},$$

где i_j – исследуемый объект.

Расстояния между объектами предполагают их представление в виде точек m -мерного пространства R^m . В этом случае могут быть использованы различные подходы к вычислению расстояний.

Рассмотренные ниже меры определяют расстояния между двумя точками, принадлежащими пространству входных переменных. Используются следующие обозначения:

$X_Q \subseteq R^m$ – множество данных, являющееся подмножеством m -мерного вещественного пространства;

$x_i = (x_{i1}, x_{i2}, \dots, x_{im}) \in X_Q, i = \overline{1, Q}$ – элементы множества данных;

$\bar{x} = \frac{1}{Q} \sum_{i=1}^Q x_i$ – среднее значение точек данных;

Евклидово расстояние. Это расстояние вычисляется следующим образом: $d_2(x_i, x_j) = \sqrt{\sum_{t=1}^m (x_{it} - x_{jt})^2}$

Расстояние по Хеммингу. Это расстояние является просто средним разностей по координатам. В большинстве случаев данная мера расстояния приводит к таким же результатам, как и для обычного расстояния Евклида, однако для нее влияние отдельных больших разностей (выбросов) уменьшается (т. к. они не возводятся в квадрат). Расстояние по Хеммингу вычисляется по формуле $d_H(x_i, x_j) = \sum_{t=1}^m |x_{it} - x_{jt}|$

Расстояние Чебышева. Это расстояние может оказаться полезным, когда желают определить два объекта как «различные», если они различаются по какой-либо одной координате (каким-либо одним измерением). Расстояние Чебышева вычисляется по формуле

$$d_\infty(x_i, x_j) = \max_{1 \leq t \leq m} |x_{it} - x_{jt}|.$$

Пиковое расстояние предполагает независимость между случайными переменными, что говорит о расстоянии в ортогональном пространстве. Но в практических приложениях эти переменные не являются независимыми:

$$d_L(x_i, x_j) = \frac{1}{m} \sum_{t=1}^m \frac{|x_{it} - x_{jt}|}{|x_{it} - x_{jt}|}.$$

Любую из приведенных мер расстояния можно выбирать с уверенностью лишь в том случае, если имеется информация о характере данных, подвергаемых кластеризации.

Так, например, пиковое расстояние предполагает независимость между случайными переменными, что говорит о расстоянии в ортогональном пространстве. Но в практических приложениях эти переменные не являются независимыми.

Результатом кластерного анализа является набор кластеров, содержащих элементы исходного множества. Кластерная модель должна описывать как сами кластеры, так и принадлежность каждого объекта к одному из них.

Для небольшого числа объектов, характеризующихся двумя переменными, результаты кластерного анализа изображают графически. Элементы представляются точками, кластеры разделяются прямыми, которые описываются линейными функциями.

2. Нейросетевое моделирование

2.1 Общая характеристика нейронных сетей

Нейронные сети – это раздел искусственного интеллекта, использующий для обработки сигналов явления, сходные с теми, которые происходят в нейронах живых организмов.

Важнейшим свойством сетей является их способность к обучению и обобщению накопленных знаний. Сеть, обученная на ограниченном множестве данных, способна обобщать полученную информацию и показывать

хорошие результаты при работе с новыми данными, которые не использовались при ее обучении.

Еще не менее важной особенностью нейронных сетей, позволяющей говорить об их широких возможностях и огромном потенциале, является параллельная обработка информации всеми звеньями, что позволяет значительно ускорить процесс.

Разнообразные способы организации нейронов между собой и способов их взаимодействия привели к созданию сетей разных типов. Каждый тип сети ассоциирован с соответствующим методом подбора весов межнейронных связей. Можно выделить следующие типы нейронных сетей: многослойный персептрон, радиальные сети RBF, сети с самоорганизацией в результате конкуренции нейронов, сети с самоорганизацией корреляционного типа, рекуррентные сети, в которых имеются сигналы обратной связи. Кроме этого, на принципах нечеткой логики работают нечеткие нейронные сети.

Гибридные сети представлены в виде объединения различных видов нейронных сетей между собой, особенно сетей с самоорганизацией и сетей, обучаемых с учителем. Первой составной частью является сеть с самоорганизацией на основе конкуренции, которая функционирует на множестве входных сигналов и группирует их в кластеры по признакам совпадения свойств (она играет роль препроцессора данных). Вторым компонентом является сеть, обучаемая с учителем, которая сопоставляет входным сигналам, отнесенным к конкретным кластерам, соответствующие им заданные значения. Данная сетевая структура делит фазу обучения на две составные части: вначале тренируется компонент с самоорганизацией, а затем - сеть с учителем. Преимуществами такого подхода является уменьшение вычислительной сложности процесса обучения и лучшая интерпретация получаемых результатов.

Использование нейронных сетей обеспечивает следующие полезные свойства систем.

Нелинейность. Существуют линейные и нелинейные искусственные нейроны. Нейронные сети, которые созданы из соединений нелинейных нейронов, являются нелинейными. В случае если входной сигнал является нелинейным (например, человеческая речь), то нелинейность является достаточно важным свойством.

Адаптивность. Нейронные сети способны приспосабливать свои синаптические веса к изменениям окружающей среды. Для работы в нестационарной среде могут быть созданы нейронные сети, изменяющие синаптические веса в реальном времени. Чем выше адаптивные возможности системы, тем более устойчивой будет ее работа в нестационарной среде. Для того чтобы использовать все достоинства адаптивности, основные параметры системы должны быть достаточно стабильными, чтобы можно было не учитывать внешние помехи, и достаточно гибкими, чтобы обеспечить реакцию на изменения среды. Эта задача обычно называется дилеммой стабильности пластичности.

Очевидность ответа. При решении задачи классификации образов можно создать нейронную сеть, которая накапливает информацию не только

для определения класса, но и для увеличения надежности принимаемого решения. В дальнейшем эта информация может использоваться для исключения сомнительных решений, что повысит продуктивность нейронной сети.

Отказоустойчивость. Нейронные сети потенциально отказоустойчивы. Следовательно, при неблагоприятных условиях их производительность падает незначительно. При повреждении какого-либо нейрона или его связи, извлечение запомненной информации затрудняется. Но повлиять на ее работоспособность могут только серьезные повреждения структуры нейронной сети. Поэтому снижение качества работы нейронной сети протекает достаточно медленно. Если же повреждение структуры является незначительным, то оно никогда не вызывает катастрофических последствий. Для гарантирования отказоустойчивости работы нейронной сети, в алгоритмы обучения необходимо закладывать соответствующие поправки.

Масштабируемость. Поскольку структура нейронных сетей является параллельной, то это потенциально ускоряет решение некоторых задач и обеспечивает масштабируемость нейронных сетей, а это предоставляет возможность представить достаточно сложное поведение с помощью иерархической структуры.

Единообразие анализа и проектирования. Нейронные сети представляют собой универсальный механизм обработки информации. Одно и то же проектное решение нейронной сети может использоваться во многих предметных областях.

Аналогия с нейробиологией. Строение нейронных сетей определяется аналогией с человеческим мозгом, который является доказательством того, что отказоустойчивые параллельные вычисления не только являются мощным инструментом решения задач.

Функции, выполняемые сетями, подразделяются на несколько групп:

- аппроксимация;
- классификация и распознавание образов;
- прогнозирование;
- идентификация и оценивание;
- ассоциативное управление.

Рассмотрим подробнее каждую из перечисленных функций.

Аппроксимирующая сеть играет роль универсального аппроксиматора функции многих переменных $y = f(x)$, который реализует нелинейную функцию. Множество задач моделирования, идентификации, обработки сигналов могут быть сформулированы в аппроксимационной постановке.

Для **классификации и распознавания образов** сеть в процессе обучения накапливает знания об основных свойствах этих образов. При обобщении концентрируются отличия образов друг от друга, которые и составляют основание для выработки классификационных решений.

В области **прогнозирования** задача сети формулируется как предсказание будущего поведения системы по имеющейся последовательности ее

предыдущих состояний. По информации о значениях временного ряда в моменты времени, предшествующие прогнозированию, сеть принимает решение о значении в текущий момент времени.

В задачах **управления** динамическими процессами нейронная сеть выполняет несколько функций: она представляет собой нелинейную модель этого процесса и определяет его основные параметры, которые нужны для выработки соответствующего управляющего сигнала, и выполняет функции следящей системы.

В задачах **ассоциации** нейронная сеть выступает в роли ассоциативного запоминающего устройства. Можно выделить память гетероассоциативного типа, с помощью которой сеть определяет взаимосвязи различных векторов и автоассоциативного типа, в которой взаимосвязности охватывают только конкретные компоненты входного вектора. Достаточно часто на вход сети поступает вектор, искаженный шумом, либо с отсутствием отдельных фрагментов данных. В этом случае сеть способна восстановить полный и очищенный от шумов исходный вектор путем генерации соответствующего ему выходного вектора.

Общая схема процесса обучения искусственной нейронной сети представлена на рисунке 2.1.

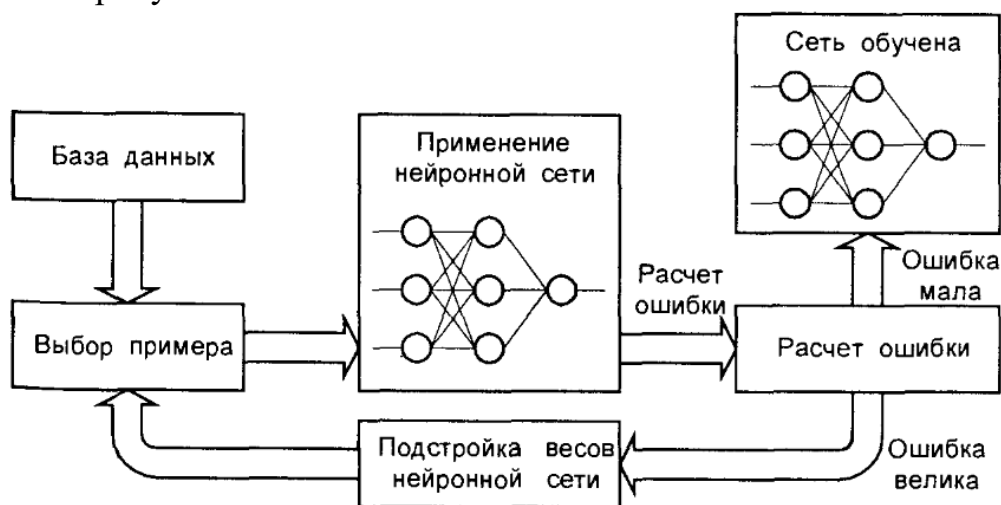


Рисунок 2.1 – Схема процесса обучения нейронной сети

Обучение с учителем

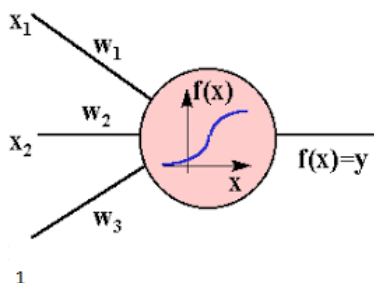
На рисунке 2.1 представлена блочная диаграмма обучения с учителем. Наличие учителя в процессе обучения рассматривается как существование знаний об окружающей среде, представленных в виде пар вход – выход. При этом сама среда неизвестна обучаемой нейронной сети. Как только обучающий вектор из окружающей среды поступает на вход, учитель, на основе имеющихся знаний, формирует и передает обучаемой нейронной сети желаемый результат, который соответствует данному входному вектору. Этот отклик представляет собой оптимальные действия, которые и должна выполнить нейронная сеть. Параметры сети изменяются с учетом обучающего

вектора и сигнала ошибки. Сигнал ошибки – это разность между желаемым сигналом и текущим откликом нейронной сети. Модификация параметров выполняется пошагово с целью имитации нейронной сетью поведения учителя. В итоге знания учителя передаются в сеть в максимально полном объеме. После окончания обучения учителя можно выключить и позволить нейронной сети работать со средой самостоятельно.

Обучение без учителя

Парадигма обучения без учителя самим названием подчеркивает отсутствие руководителя, контролирующего процесс настройки весовых коэффициентов. При использовании такого подхода не существует маркированных примеров, по которым проводится обучение сети. Обучение без учителя (или обучение на основе самоорганизации) осуществляется без вмешательства внешнего учителя, или корректора, контролирующего процесс обучения. Существует лишь независимая от задачи мера качества представления, которому должна научиться нейронная сеть, и свободные параметры сети оптимизируются по отношению к этой мере. После обучения сети она способна формировать внутреннее представление кодируемых признаков входных данных и, таким образом, автоматически создавать новые классы. Примером сети обучаемой без учителя является сеть Кохонена.

Основным элементом нейронной сети является искусственный нейрон. Он выполняет операцию нелинейного преобразования суммы произведений входных сигналов на весовые коэффициенты:



$$y = f(w_3 + x_2 w_2 + x_1 w_1)$$

Оператор нелинейного преобразования называется функцией активации нейронного элемента. В качестве оператора нелинейного преобразования могут использоваться различные функции, которые определяются в соответствии с решаемой задачей и типом нейронной сети.

Пороговая функция

В качестве пороговой функции активации может использоваться или биполярная, или бинарная пороговая функция.

Если применяется пороговая биполярная функция активации, то выходное значение нейронного элемента вычисляется как

$$f(s) = \begin{cases} 1, & s > 0 \\ -1, & s \leq 0 \end{cases}$$

Основным недостатком пороговых функций активации является их недифференцируемость.

Линейная ограниченная функция

В этом случае выходное значение нейрона определяется следующим образом

$$f(s) = \begin{cases} p, & s > \alpha \\ -p, & s < -\alpha \\ s, & -\alpha \leq s \leq \alpha \end{cases}$$

Модифицированная пороговая функция

Данная пороговая функция используется в двунаправленной ассоциативной памяти. Выходное значение нейронного элемента характеризуется следующей функцией:

$$y(t+1) = \begin{cases} 1, & s > 0 \\ y(t), & s = 0 \\ -1, & s < 0 \end{cases}$$

здесь t – параметр времени.

Сигмоидная функция

Эта функция является непрерывным аналогом бинарной пороговой функции активации и представляет собой непрерывную, возрастающую, ограниченную функцию в диапазоне значений $[0, 1]$:

$$y = \frac{1}{1 + e^{-cs}}$$

где $c > 0$ – коэффициент, характеризующий ширину сигмоидной функции по оси абсцисс.

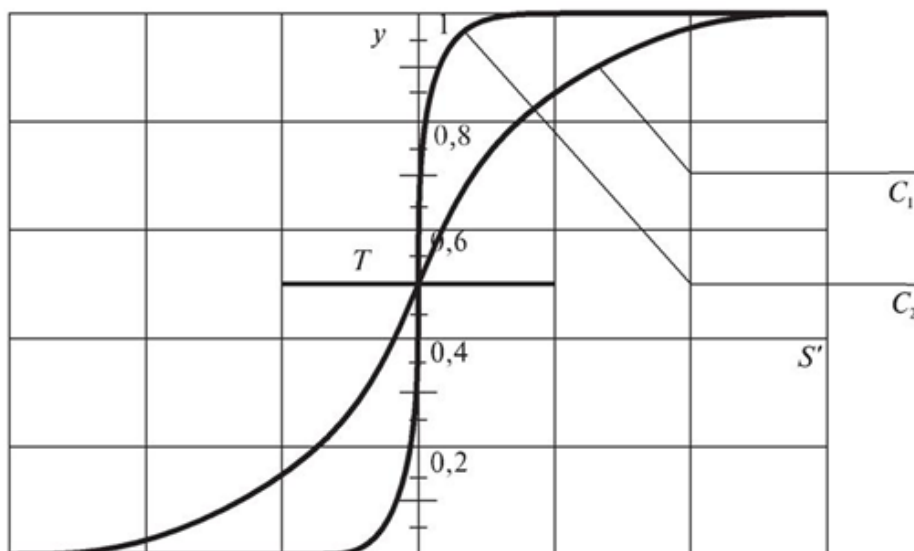


Рисунок 2.2 – Сигмоидная функция

Сигмоидная функция монотонна и всюду дифференцируема, поэтому она получила широкое распространение в искусственных нейронных сетях.

Гиперболический тангенс

Она определяется следующим образом:

$$y = \frac{e^{cs} - e^{-cs}}{e^{cs} + e^{-cs}}$$

где c , как и в случае с сигмоидной функцией, характеризует ширину функции активации гиперболического тангенса по оси абсцисс.

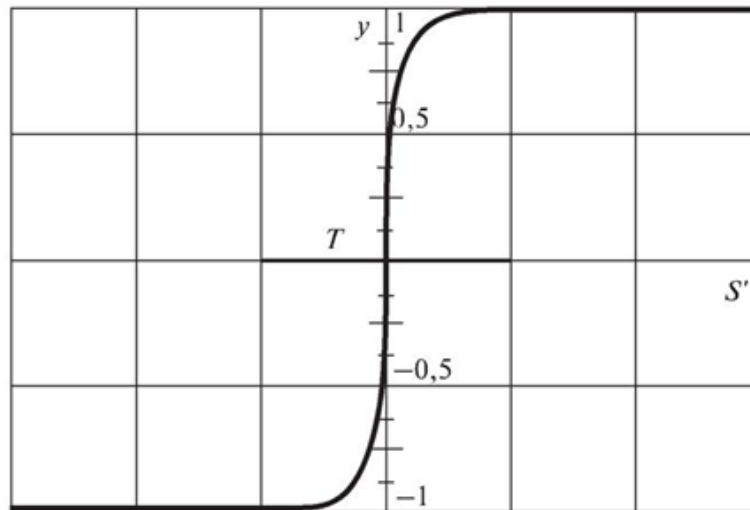


Рисунок 2.3 – Гиперболический тангенс

Радиально-базисная функция

Эта функция определяется функцией Гаусса для нормального закона распределения. Согласно закону

$$y = \exp\left(\frac{-s^2}{2\sigma^2}\right)$$

где σ – среднеквадратичное отклонение ширины радиально-базисной функции.

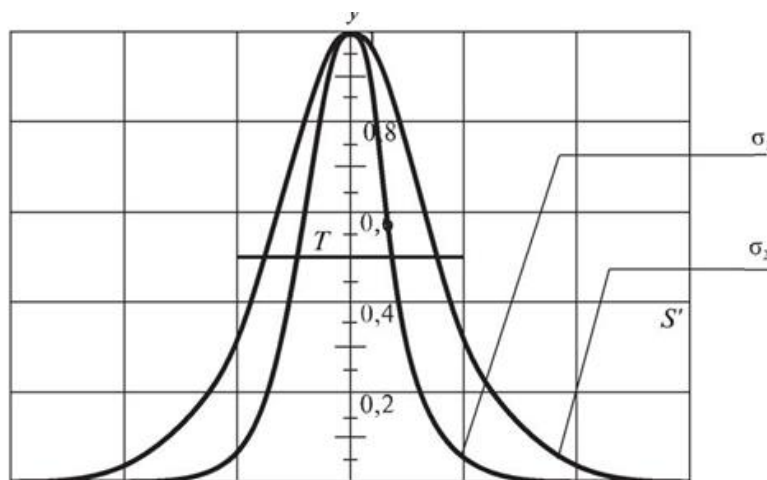


Рисунок 2.4 – Радиально-базисная функция

Функция активации softmax

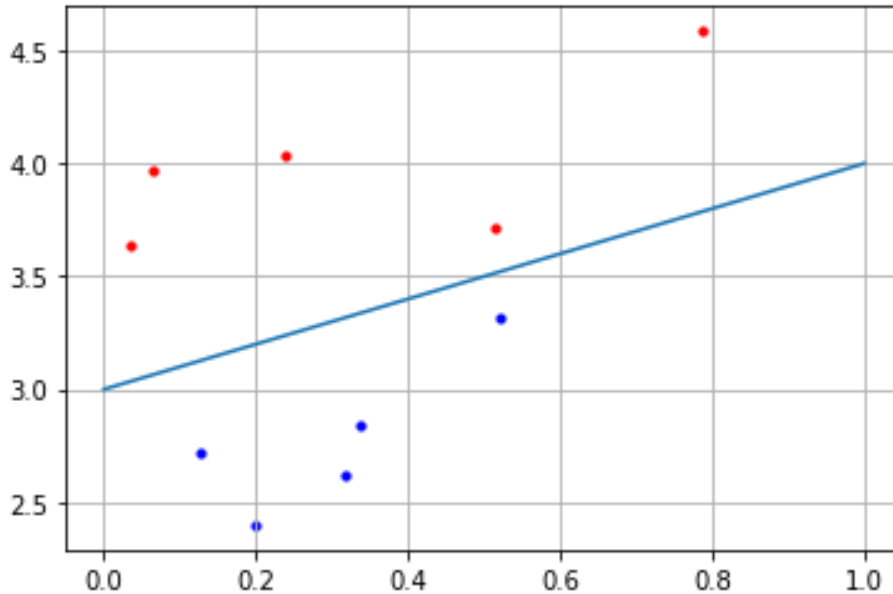
Данная функция активации применяется обычно в последнем слое нейронной сети и используется в системах распознавания образов. Так, выходное значение j -го нейронного элемента в соответствии с данной функцией активации определяется следующим образом:

$$y_j = \frac{e^{s_j}}{\sum_j e^{s_j}}$$

В следующем примере строится разделяющая прямая с помощью одного нейрона, которая разбивает все объекты на 2 группы.

```
import numpy as np
import matplotlib.pyplot as plt
N = 5
b = 3
x1 = np.random.random(N)
x2 = x1 + [np.random.randint(10)/10 for i in range(N)] + b
C1 = [x1, x2]
x1 = np.random.random(N)
x2 = x1 - [np.random.randint(10)/10 for i in range(N)] - 0.1 + b
C2 = [x1, x2]
f = [0+b, 1+b]
w2 = 0.5
w3 = -b*w2
w = np.array([-w2, w2, w3])
for i in range(N):
    x = np.array([C1[0][i], C1[1][i], 1])
    y = np.dot(w, x)
    if y >= 0:
        print("Класс C1")
    else:
        print("Класс C2")

plt.scatter(C1[0][:], C1[1][:], s=10, c='red')
plt.scatter(C2[0][:], C2[1][:], s=10, c='blue')
plt.plot(f)
plt.grid(True)
plt.show()
```



2.2 Обучение нейронных сетей

Последовательное обучение

Последовательным называется обучение, при котором модификация синаптических связей сети происходит после подачи каждого образа из обучающей выборки на нейронную сеть. В этом случае в методе градиентного спуска используется квадратичная ошибка нейронной сети для одного любого входного образа:

$$E = \frac{1}{2} \sum_{j=1}^m (y_j - a_j)^2$$

Тогда в соответствии с методом градиентного спуска для минимизации суммарной квадратичной ошибки сети весовые коэффициенты и пороги нейронной сети необходимо изменять с течением времени по следующим выражениям:

$$w_{ij}(t + 1) = w_{ij}(t) - \alpha \frac{\partial E}{\partial w_{ij}(t)}$$

α – шаг обучения.

Проведя вычисления, получим выражения для обучения нейронной сети, которые называются дельта правилом:

$$w_{ij}(t + 1) = w_{ij}(t) - \alpha (y_j - a_j) x_i$$

Алгоритм обучения однослойного персептрона, в основе которого лежит дельта-правило, состоит из следующих шагов:

1. задается скорость обучения α ($0 < \alpha < 1$) и желаемая квадратичная ошибка сети, которой необходимо достичь в процессе обучения;
2. случайным образом инициализируются синаптические связи нейронной сети в достаточно узком диапазоне значений, например $[0, 1]$.

3. последовательно подаются входные образы из обучающей выборки на нейронную сеть и для каждого образа производятся следующие действия:
 - вычисляются выходные значения сети;
 - производится изменение весовых коэффициентов нейронной сети;
4. алгоритм продолжается до тех пор, пока суммарная квадратичная ошибка сети не станет меньше заданной.

Другой критерий – остановка окончания алгоритма: алгоритм продолжается, пока не перестанут изменяться синаптические связи или уменьшаться значение суммарной квадратичной ошибки сети.

Групповое обучение

Групповым называется обучение, при котором модификация синаптических связей происходит после подачи группы или всех образов из обучающей выборки на нейронную сеть. В этом случае в методе градиентного спуска используется суммарная квадратичная ошибка для группы из L_1 – образов, после подачи которых на вход сети модифицируются синаптические связи:

$$w_{ij}(L_1) = w_{ij}(0) - \alpha \frac{\partial E_s(L_1)}{\partial w_{ij}}$$

здесь $w_{ij}(0)$ – синаптические связи до подачи образов для обучения, $E_s(L_1)$ – суммарная квадратичная ошибка:

$$E_s(L_1) = \frac{1}{2} \sum_{k=1}^{L_1} \sum_{j=1}^m (y_j^k - a_j^k)^2$$

$$y_j^k = \sum_{i=1}^n x_i^k w_{ij}$$

В случае группового обучения весовые коэффициенты и пороговые значения должны изменяться следующим образом:

$$w_{ij}(L_1) = w_{ij}(0) - \alpha(t) \frac{1}{2} \sum_{k=1}^{L_1} (y_j^k - a_j^k) x_i^k$$

Данные выражения минимизируют суммарную квадратичную ошибку сети для всех образов из обучающей выборки.

Простейшая НС – персептрон, представляет собой упрощенное отражение работы биологической сети, состоящей из нейронов, соединенных между собой дендридами и аксонами:

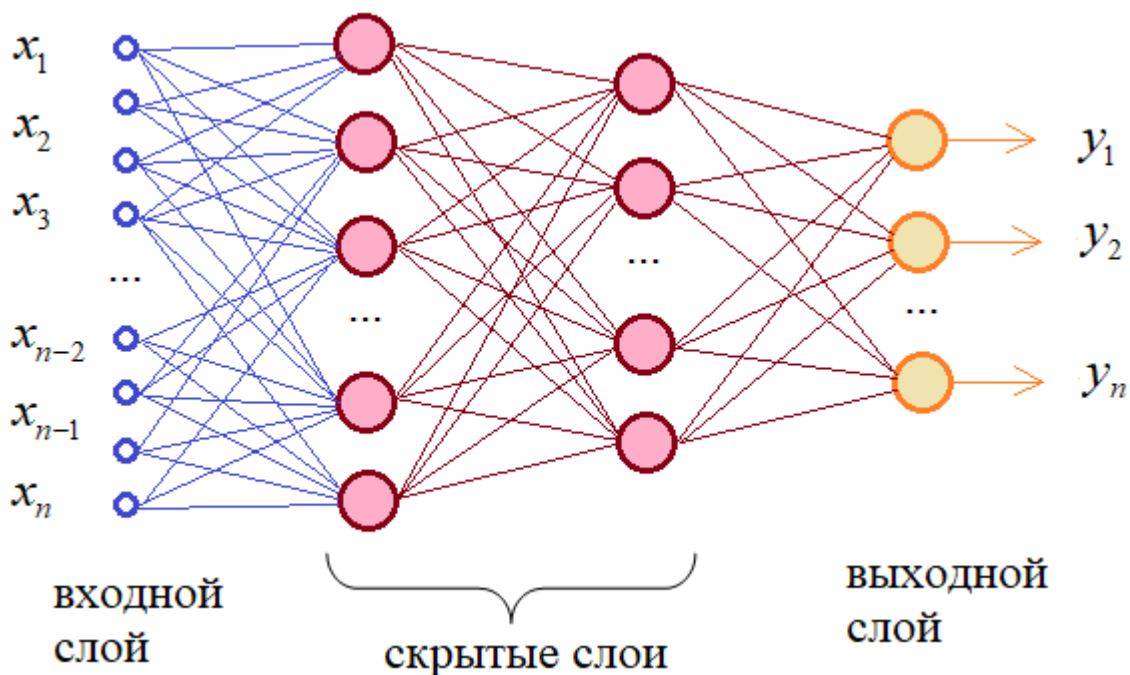


Рисунок 2.5 – Нейронная сеть

$X = (X_1, \dots, X_k)$ – входные вектора

$A = (A_1, \dots, A_k)$ – правильные выходные вектора

(X, A) – обучающая выборка

W – вектор весов обучающей выборки

$N(W, X)$ – функция, соответствующая нейронной сети

$Y = N(W, X)$ – ответ нейронной сети

$D(Y, A) = \sum_{j=1}^m (Y[j] - A[j])^2$ – функция ошибки

$D_i(Y) = D(Y, A_i)$ – функция ошибки на i -ом примере

$E_i(W) = D_i(N(W, X_i))$ – ошибка сети на i -ом примере

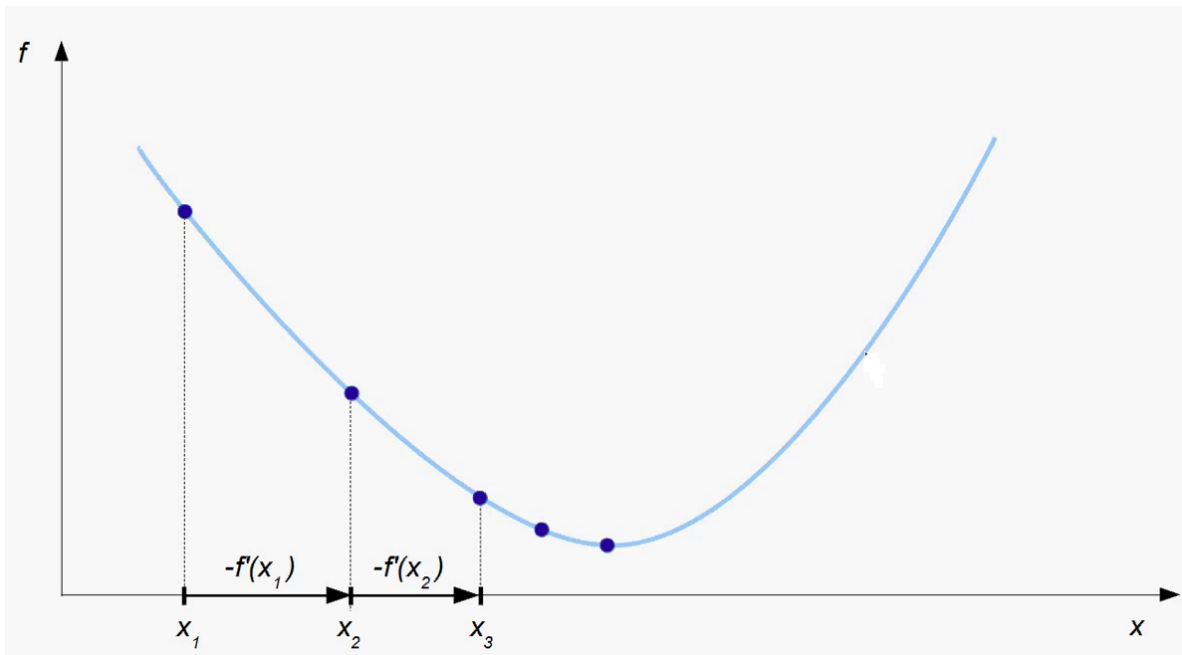
$E(W) = \sum_{i=1}^k E_i(W)$ – ошибка сети на всей обучающей выборки

Задачи состоит в нахождении вектора W , такого, что $E(W) \rightarrow \min$ (обучение на всей выборке) и

$E_i(W) \rightarrow \min$ (обучение на одном примере)

Представим на рисунке функцию одной переменной $f(x)$. Производная функции в точке x_1 это число, которое показывает возрастает функция или убывает. В точке x_1 функция убывает, значит ее производная со знаком “-” положительна. Затем мы сместимся на эту производную, т.е. к x_1 прибавим величину производной. Таким образом, мы из этой точки сделаем шаг в направлении обратном направлению производной. В результате мы переместились в точку x_2 , значение функции $f(x_2) < f(x_1)$. Продолжая смещаться на величину производной на каждом шаге функция будет убывать, а значит в самом конце она достигнет минимума.

Таким образом, если мы сдвигаемся в направлении обратном направлению производной, то мы уменьшаем значение нашей функции.



Алгоритм градиентного спуска для функции одной переменной.

1. Инициализировать x_1 случайным значением
2. $i = 1$
3. $x_{i+1} = x_i - \epsilon f'(x_i)$
4. $i++$
5. если $f(x_i) - f(x_{i+1}) > \epsilon$ перейти к пункту 3.

Для случая нескольких переменных алгоритм будет выглядеть следующим образом.

1. Инициализировать W_1 случайным значением из \mathbb{R}^n
2. $i = 1$
3. $W_{i+1} = W_i - \epsilon \nabla f(W_i)$
4. $i++$
5. если $f(W_i) - f(W_{i+1}) > \epsilon$ перейти к пункту 3.

Частная производная и градиент.

Пусть $f(x_1, \dots, x_n)$ – функция n переменных. Частная производная по i -ой переменной:

$$\frac{\partial f}{\partial x_i}(x_1, \dots, x_n) = \lim_{\epsilon \rightarrow 0} [f(x_1, x_2, \dots, x_i + \epsilon, \dots, x_n) - f(x_1, x_2, \dots, x_i, \dots, x_n)] / \epsilon;$$

$$\frac{\partial f}{\partial x_i}: \mathbb{R}^n \rightarrow \mathbb{R}.$$

$$\text{Градиент функции: } \nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right)$$

$$\nabla f: \mathbb{R}^n \rightarrow \mathbb{R}^n.$$

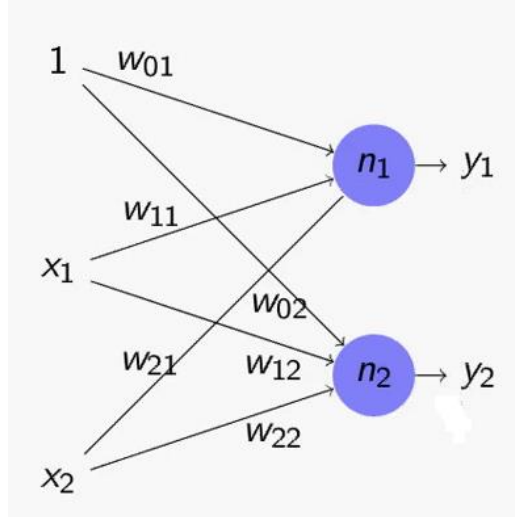
Производная сложной функции:

Рассмотрим функцию $f = f(x_1, \dots, x_n)$, где $x_i = x_i(y_1, \dots, y_m)$.

Тогда $f(y_1, \dots, y_m) = f(x_1(y_1, \dots, y_m), \dots, x_n(y_1, \dots, y_m))$.

$$\frac{\partial f}{\partial y_i} = \sum_{j=1}^n \frac{\partial f}{\partial x_j} \frac{\partial x_j}{\partial y_i}$$

Обратное распространение ошибки.

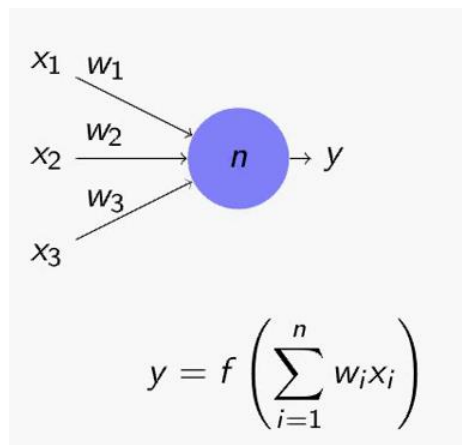


$$D_k(y_1, y_2) = (y_1 - a_1)^2 + (y_2 - a_2)^2$$

$$\frac{\partial D_k}{\partial y_1} = 2(y_1 - a_1)$$

$$\frac{\partial D_k}{\partial y_2} = 2(y_2 - a_2)$$

Функция y – это функция активации от взвешенной суммы входов



$$y = f \left(\sum_{i=1}^n w_i x_i \right)$$

$$y_1 = f(w_{01} + x_1 w_{11} + x_2 w_{21})$$

Обозначим $s_1 = w_{01} + x_1 w_{11} + x_2 w_{21}$

$$\text{Тогда } \frac{\partial y_1}{\partial w_{21}} = f'(s_1) x_2$$

$$y_2 = f(w_{02} + x_1 w_{12} + x_2 w_{22})$$

Обозначим $s_2 = w_{02} + x_1 w_{12} + x_2 w_{22}$

Тогда $\frac{\partial y_2}{\partial w_{21}} = 0$

Вычислим частную производную функции E_k по этому весу.

$$E_k(W) = D_k(y_1(w_{01}, w_{11}, w_{21}), y_2(w_{02}, w_{12}, w_{22}))$$

$$\frac{\partial E_k}{\partial w_{21}} = \frac{\partial D_k}{\partial y_1} \frac{\partial y_1}{\partial w_{21}} + \frac{\partial D_k}{\partial y_2} \frac{\partial y_2}{\partial w_{21}} = 2(y_1 - a_1) f'(s_1) x_2$$

По одному весу мы вычислили производную, по остальным вычисляется аналогично.

Обобщим для общего случая.

$$D_k(y_1, \dots, y_n) = (y_1 - a_1)^2 + \dots + (y_n - a_n)^2$$

$$\frac{\partial D_k}{\partial y_i} = 2(y_i - a_i)$$

Частичная сумма внутри функции активации записывается по формуле:

$$s_i = \sum_{j=0}^m x_j w_{ji}$$

Выходной нейрон

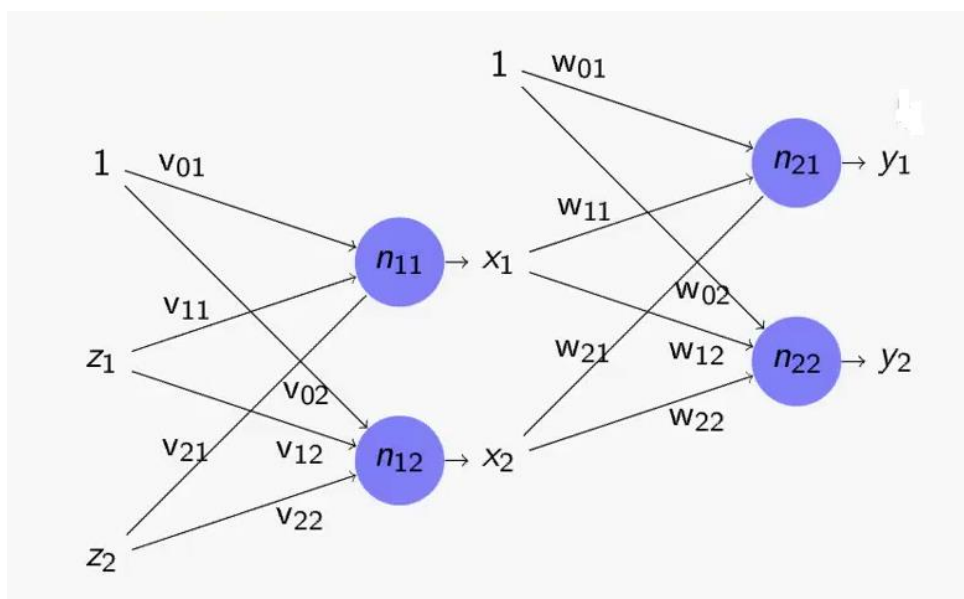
$$y_i = f(s_i)$$

Тогда $\frac{\partial y_i}{\partial w_{ji}} = f'(s_i) x_j$

$$E_k(W) = D_k(y_1(w_{01}, \dots, w_{m1}), \dots, y_n(w_{0n}, \dots, w_{mn}))$$

$$\frac{\partial E_k}{\partial w_{ji}} = \sum_{l=1}^n \frac{\partial D_k}{\partial y_l} \frac{\partial y_l}{\partial w_{ji}} = 2(y_l - a_l) f'(s_l) x_j$$

Рассмотрим градиент ошибки для случая, когда сеть состоит из нескольких слоев.



Входные нейроны второго слоя получены как результат работы первого слоя и при их выработки использовались веса первого слоя.

$$E_k(W) = D_k(y_1, \dots, y_n)$$

$$y_i = y_i(x_1, \dots, x_m)$$

$$x_j = x_j(v_{0j}, \dots, v_{rj})$$

Функция ошибки

$$D_k(y_1, \dots, y_n) = (y_1 - a_1)^2 + \dots + (y_n - a_n)^2$$

Ее частная производная по y

$$\frac{\partial D_k}{\partial y_i} = 2(y_i - a_i)$$

Взвешенная сумма

$$s_i = \sum_{j=0}^m x_j w_{ji}$$

$$y_i = f(s_i)$$

Тогда $\frac{\partial y_i}{\partial x_j} = f'(s_i) w_{ji}$

$$\frac{\partial D_k}{\partial x_j} = \sum_{i=1}^n \frac{\partial D_k}{\partial y_i} \frac{\partial y_i}{\partial x_j} = 2 \sum_{i=1}^n (y_i - a_i) f'(s_i) w_{ji}$$

Мы получили производную ошибки по входам нейронной сети.

$$\frac{\partial E_k}{\partial v_{rs}} = \sum_{j=1}^m \frac{\partial D_k}{\partial x_j} \frac{\partial x_j}{\partial v_{rs}}$$

$$\frac{\partial D_k}{\partial z_l} = \sum_{j=1}^m \frac{\partial D_k}{\partial x_j} \frac{\partial x_j}{\partial z_l}$$

Для минимизации суммарной квадратичной ошибки сети при последовательном обучении весовые коэффициенты элементов любых двух слоев i и j нейронной сети с течением времени должны изменяться следующим образом:

$$w_{ij}(t + 1) = w_{ij}(t) - \alpha \gamma_j f'(s_j) y_i$$

γ_j – ошибка j -ого нейрона

Для любого скрытого слоя i ошибка i -го нейронного элемента определяется рекурсивным образом через ошибки нейронов следующего j -го слоя:

$$\gamma_i = \sum_{j=1}^m \gamma_j f'(s_j) w_{ij}$$

где m – количество нейронов следующего слоя по отношению к слою i ;

w_{ij} – синаптическая связь между i -м и j -м нейроном различных слоев;

S_j – взвешенная сумма j -го нейрона.

В случае группового обучения модификация синаптических связей происходит после подачи группы или всех образов из обучающей выборки на нейронную сеть. В этом случае в методе градиентного спуска используется суммарная квадратичная ошибка для группы из L_1 образов, после подачи которых на вход сети модифицируются синаптические связи:

$$w_{ij}(t + 1) = w_{ij}(t) - \alpha \sum_{k=1}^{L_1} \gamma_j^k f'(s_j^k) y_i^k$$

Выходное значение j -го нейронного элемента для k -го образа

$$y_j^k = f(s_j^k)$$

Лабораторная работа.

Даны временные ряды, число уровней временного ряда равно 11. С помощью многослойного персептрона выполнить прогнозирование на один уровень.

1.

t	1	2	3	4	5	6	7	8	9	10	11
Объем операций, млрд руб.	75	60	65	40	43	28	20	22	16	20	18

2.

t	1	2	3	4	5	6	7	8	9	10	11
Объем ВВП, млрд долл.	5,0	5,3	6,2	7,0	7,5	8,2	9,0	9,5	10,3	11,0	11,2

3.

t	1	2	3	4	5	6	7	8	9	10	11
Выручка, млн руб.	95	63	50	32	15	20	28	53	70	90	95

4.

T	1	2	3	4	5	6	7	8	9	10	11
Объем операций, млн руб.	5	25	40	50	55	60	61	62	63	64	65

5.

T	1	2	3	4	5	6	7	8	9	10	11
Цена, долл/барр	16,1	18,52	18,23	12,21	17,25	26,2	22,81	23,74	26,78	33,64	34,05

6.

t	1	2	3	4	5	6	7	8	9	10	11
Экспорт	171,6	164,07	165,74	186,17	185,67	184,16	186,18	186,42	204,4	232,76	233,54

7.

T	1	2	3	4	5	6	7	8	9	10	11
Объем рынка, млн долл.	15,1	17,53	18,43	11,21	16,25	25,2	21,81	24,74	27,78	33,64	34,05

8.

Т	1	2	3	4	5	6	7	8	9	10	11
Объем продаж, млн руб.	95	95	85	11,21	75	70	85	83	82	83	85

9.

Т	1	2	3	4	5	6	7	8	9	10	11
Объем рынка, млн долл.	17,92	20,67	19,99	19,72	23,97	28,50	24,44	25,02	28,83	27,27	29,05

10.

Т	1	2	3	4	5	6	7	8	9	10	11
Объем продаж, млн руб.	80	79	85	80	75	68	65	68	65	62	61

11.

Т	1	2	3	4	5	6	7	8	9	10	11
Объем экспорта, млн тонн	129,7	116,0	115,8	116,8	127,2	124,6	132,5	162,1	174,3	208,5	201,3

12.

Т	1	2	3	4	5	6	7	8	9	10	11
Цена, долл/барр	26,1	28,52	28,23	22,21	27,25	36,2	32,81	33,74	36,78	43,64	44,05

3. Анализ временных рядов и прогнозирование

3.1 Постановка задачи

Временной ряд – это последовательность значений, описывающих протекающий во времени процесс, измеренных в последовательные моменты времени, обычно через равные промежутки.

Исследование скорости и интенсивности развития временных рядов часто не позволяет сразу определить основную тенденцию поступательного движения изучаемого явления. Это зависит от того, что уровни временного ряда со временем меняются, колеблются, и эти колебания не одинаковы и могут быть вызваны следующими причинами:

- влиянием общих факторов, определяющих главное направление, основную тенденцию развития явления;
- влиянием факторов общего характера, действующих периодически, сезонных колебаний и т.д.;
- влиянием специфических факторов, каждый из которых действует в разных направлениях, и их действие несущественно с точки зрения результатов развития явления, случайных колебаний.

Тип связи между компонентами временного ряда можно определить по нормальному распределению отклонений эмпирических значений уровней временного ряда от теоретических, полученных по уравнению тренда.

В случае нормальности распределения абсолютных отклонений связь является аддитивной, а относительных — мультипликативной.

Основные компоненты могут воздействовать на величину уровней временного ряда по-разному:

если факторы, образующие эти компоненты, мультипликативные, то значения уровней временных рядов являются произведением этих компонентов:

$$\bar{y}_t = T * C * \varepsilon,$$

где T – тенденция, C – сезонный компонент, ε – случайный компонент.

если факторы аддитивные, то значения уровней временных рядов являются суммой компонентов:

$$\bar{y}_t = T + C + \varepsilon,$$

если факторы временного ряда выражены комбинированно, то значения уровней являются или произведением, или суммой компонентов:

$$\bar{y}_t = T * C + \varepsilon.$$

Тип связи между компонентами можно также определить по динамике отклонений эмпирических значений уровней временного ряда от теоретических, полученных по уравнению тренда. Если абсолютные отклонения имеют тенденцию к росту, а относительные варьируют приблизительно на одинаковом уровне, то это свидетельствует о мультипликативной связи тренда и сезонного компонента.

На практике выделить компоненты сложно, так как отдельные последующие значения временных рядов зависят от предыдущих. Поэтому неверно допускать, что факторы, влияющие на колебания уровней, независимы. Кроме того, статистическая совокупность, изучаемая в течение длительного периода, перестает быть такой же самой совокупностью, так как могут измениться основные факторы, влияющие на ее формирование.

Тренд – это аналитическая функция, которая связывает единым законом движения все последовательные уровни временного ряда. Тренд описывает общую тенденцию на базе лишь одного фактора – фактора времени.

Для моделирования тенденции целесообразно воспользоваться методом наименьших квадратов.

Моделирование тенденции.

Анализ и моделирование тенденции временного ряда целесообразно начинать с выявления наличия тенденции в целом. Для этой цели наиболее эффективны и дают хорошие результаты такие методы как кумулятивный T -критерий и критерий знаков разностей отклонений Валлиса и Мура.

Кумулятивный T -критерий позволяет определить наличие не только самой тенденции, но и ее математического выражения – тренда. Выдвигается основная гипотеза (Н₀:) об отсутствии тенденции в исходном временном ряду. Расчетное значение критерия определяется как отношение

накопленной суммы квадратов отклонений эмпирических значений уровней временного ряда от их среднего значения и самих отклонений по формуле:

$$T_p = \frac{\sum_n z_n^2}{\sigma_y^2}$$

$z_n = \sum (y_t - \bar{y})$ – накопленный итог отклонений эмпирических значений от среднего уровня исходного временного ряда

$$\sigma_y^2 = \sum_n y_t^2 - \bar{y}^2 * n$$

Если анализируется достаточно длинный временной ряд, то для расчета значений критерия можно использовать нормированное отклонение:

$$t_p = \frac{T_p - \frac{n+1}{6}}{\sqrt{(n-2) \frac{2n-1}{90}}}$$

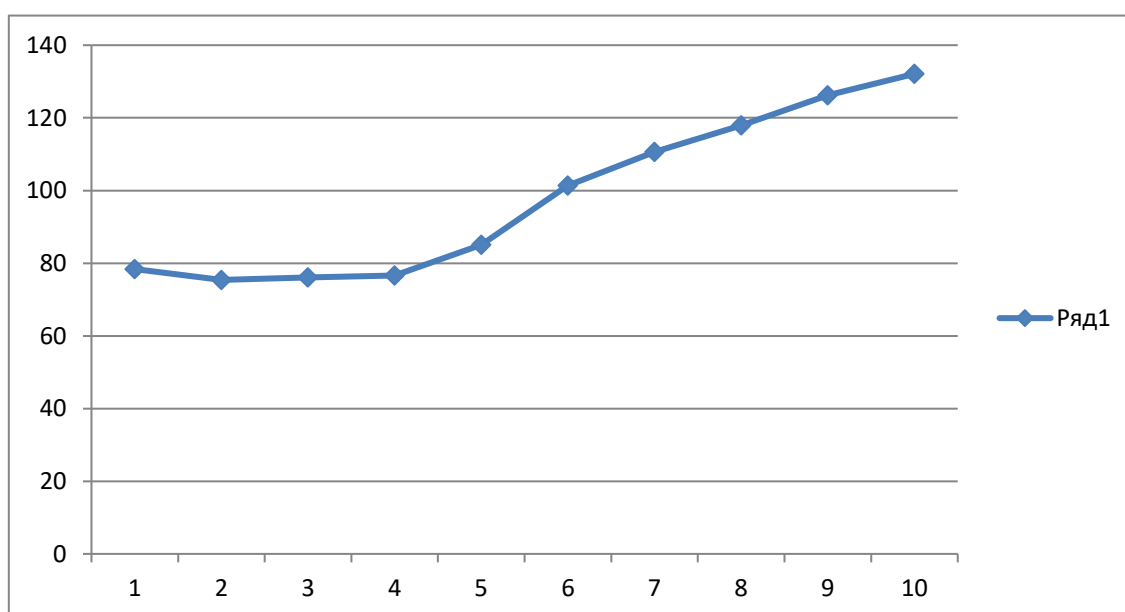
Расчетные значения кумулятивного Т-критерия и t_p сравниваются с критическими при заданном уровне значимости α . Если $T_p > T_{кр}$, то гипотеза об отсутствии тенденции отвергается, следовательно, в исходном временном ряду существует тенденция, описываемая трендом. В противном случае, если $T_p < T_{кр}$ или $t_p < t_{кр}$, признается отсутствие тенденции в ряду динамики.

Пример.

Дан временной ряд.

2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
78,4	75,4	76,1	76,6	85,1	101,4	110,6	117,9	126,2	132,1

Определить наличие тенденции в данном временном ряду.



t	y	Y ²	сред- нее	y-среднее	Z _n	Z _n ²
1	78,4	6146,56	97,98	-19,58	-19,58	383,3764
2	75,4	5685,16		-22,58	-42,16	1777,466
3	76,1	5791,21		-21,88	-64,04	4101,122
4	76,6	5867,56		-21,38	-85,42	7296,576
5	85,1	7242,01		-12,88	-98,3	9662,89
6	101,4	10281,96		3,42	-94,88	9002,214
7	110,6	12232,36		12,62	-82,26	6766,708
8	117,9	13900,41		19,92	-62,34	3886,276
9	126,2	15926,44		28,22	-34,12	1164,174
10	132,1	17450,41		34,12	0	0
Сумма	979,8	100524,08				44040,80

$$\sigma^2 = 4523,276$$

$$T_p = 9,736483$$

Критические значения кумулятивного T-критерия

n	Для проверки существенности тренда	
	T _{кр}	t _{кр}
6	2,62	2,11
7	3,11	2,10
8	3,59	2,09
9	4,07	2,09
10	4,55	2,09
11	5,02	2,08
12	5,49	2,08
13	5,96	2,07
14	6,42	2,07
15	6,89	2,06
16	7,36	2,06
17	7,82	2,06
18	8,29	2,05
19	8,76	2,05
20	9,22	2,04

$T_p(9,736) > T_{кр}(4,55)$ гипотеза об отсутствии тренда отвергается

Фазочастотный критерий знаков разностей Валлиса и Мура также позволяет определить наличие основной тенденции в целом во временном ряду и относится к группе непараметрических методов оценивания тенденции.

Выдвигается основная гипотеза (H₀:) о том, что знаки разностей между (yⁱ⁺¹-yⁱ) образуют случайную последовательность. При этом формируется понятие фазы как последовательности одинаковых знаков разностей.

Если знаки при определении числа фаз (без 1-ой и последней) образуют случайную последовательность, то расчетное значение критерия определяется по формуле:

$$t_p = \frac{\left| h - \frac{2n-7}{3} \right| - 0.5}{\sqrt{\frac{16n-29}{90}}}$$

где n - число уровней ряда, h - число фаз.

Если расчетное значение (t_p) критерия превысит критическое (табличное) $t_{кр}$ при заданном уровне значимости α , то гипотеза H^0 : о том, что знаки разностей между последовательными уровнями временного ряда образуют случайную последовательность, отвергается. Следовательно уровни исходного временного ряда не образуют случайную последовательность, а имеют определенную закономерность в их изменении, а следовательно, и в ряду существует тенденция.

Проверка на наличие тенденции среднего уровня и дисперсии может быть произведена **методом сравнения средних уровней временного ряда и методом Фостера–Стюарта**.

Метод сравнения средних уровней временного ряда предполагает, что исходный временной ряд разбивается на две приблизительно равные части по числу членов ряда, каждая из которых рассматривается как самостоятельная, независимая выборочная совокупность, имеющая нормальное распределение. При этом решаются две задачи.

I. Если временной ряд имеет тенденцию, то средние, вычисленные для каждой совокупности в отдельности, должны существенно, значимо различаться между собой. Если же расхождение незначимо, несущественно и носит случайный характер, то временной ряд не имеет тенденции средней.

Таким образом, проверка гипотезы (H^0 :) о наличии тенденции в исследуемом ряду сводится к проверке гипотезы о равенстве средних двух нормально распределенных совокупностей, то есть:

$$H^0: \bar{y}_1 = \bar{y}_2$$

$$H^1: \bar{y}_1 \neq \bar{y}_2$$

Гипотеза проверяется на основе t -критерия Стьюдента, расчетное значение которого определяется по формуле:

$$t_p = \frac{\bar{y}_1 - \bar{y}_2}{\sqrt{(n_1 - 1)\sigma_1^2 + (n_2 - 1)\sigma_2^2}} \sqrt{\frac{n_1 n_2 (n_1 + n_2 - 2)}{n_1 + n_2}}$$

\bar{y}_1, \bar{y}_2 - средние уровни временного ряда согласно порядка разбиения;
 n_1, n_2 - число уровней временного ряда, соответственно первой и второй части;

σ_1^2, σ_2^2 - дисперсия уровней ряда.

Расчетное значение (t_p) критерия сравнивается с его критическим (табличным) значением ($t_{кр}$) при уровне значимости α и числе степеней свободы $v = n - 2$.

Если $t_p > t_{кр}$, то гипотеза о равенстве средних уровней двух нормально распределенных совокупностей отвергается, следовательно расхождение между вычисленными средними значимо, существенно и носит неслучайный характер, и, следовательно, во временном ряду существует тенденция средней и существует тренд.

II. Если временной ряд имеет тенденцию, то дисперсии, вычисленные для каждой совокупности в отдельности, должны существенно и значимо различаться между собой. Если же расхождение между ними не значимо, то временной ряд не имеет тенденции дисперсии. Таким образом проверяется гипотеза (H_0 :) об отсутствии тенденции в дисперсиях в исходном временном ряду, которая сводится к проверке гипотезы о равенстве дисперсий двух нормально распределенных совокупностей, то есть:

$$H^0: \sigma_1^2 = \sigma_2^2$$

$$H^1: \sigma_1^2 \neq \sigma_2^2$$

Гипотеза проверяется на основе F-критерия Фишера-Снедекора, расчетное значение которого определяется по формуле:

$$F_p = \frac{\sigma_2^2}{\sigma_1^2}, \text{ если } \sigma_2^2 > \sigma_1^2$$

$$F_p = \frac{\sigma_1^2}{\sigma_2^2}, \text{ если } \sigma_1^2 > \sigma_2^2$$

Проверка гипотезы осуществляется на основе сравнения расчетного и критического значений F-критерия, полученного при заданном уровне значимости α и числе степеней свободы v_1 и v_2 :

$$\text{если } \sigma_2^2 > \sigma_1^2, \text{ то } v_1 = n_2 - 1, v_2 = n_1 - 1,$$

$$\text{если } \sigma_1^2 > \sigma_2^2, \text{ то } v_1 = n_1 - 1, v_2 = n_2 - 1.$$

Гипотеза о равенстве дисперсий двух нормально распределенных совокупностей отвергается, если $F_p > F_{кр}$. Следовательно, расхождение между вычисленными дисперсиями значимо, существенно, носит неслучайный характер и в ряду динамики существует тенденция в дисперсиях и существует тренд.

Данный метод дает вполне приемлемые результаты лишь в случае рядов с монотонной тенденцией. Если же ряд динамики меняет общее направление развития, то точка поворота тенденции может оказаться близкой к середине ряда, в силу этого средние двух отрезков ряда будут близки и проверка может не показать наличие тенденции.

3.2 Использование метода скользящей средней в прогнозировании

Метод скользящей средней – один из методов статистического прогнозирования.

Необходимость применения скользящей средней вызывается следующими обстоятельствами. Бывают случаи, когда имеющиеся данные динамического ряда не позволяют обнаруживать какую-либо тенденцию развития (тренд) того или иного процесса (из-за случайных и периодических колебаний исходных данных). В таких случаях для лучшего выявления тенденции прибегают к методу скользящей средней.

Экстраполяция по скользящей средней может применяться для целей краткосрочного прогнозирования. Метод скользящей средней состоит в замене фактических уровней динамического ряда расчетными, которые имеют значительно меньшие колебания, чем исходные данные. При этом средняя рассчитывается по группам данных за определенный интервал времени, причем каждая последующая группа образуется со сдвигом на один год (месяц). В результате подобной операции первоначальные колебания динамического ряда сглаживаются, поэтому и операция называется сглаживанием рядов динамики (основная тенденция развития выражается при этом уже в виде некоторой плавной линии).

Метод скользящей средней называется так потому, что при вычислении средние как бы скользят от одного периода к другому; с каждым новым шагом средняя как бы обновляется, впитывая в себя новую информацию о фактически реализуемом процессе. Таким образом, при прогнозировании исходят из простого предположения, что следующий во времени показатель по своей величине будет равен средней, рассчитанной за последний интервал времени.

Пример. Количество сбоев в информационной системе торговой компании составило:

в январе -60,
в феврале -85,
в марте-80,
в апреле-92,
в мае-88,
в июне-96,

тогда прогноз сбоев на июль (для 5-ти месячного периода) составит:
 $(85 + 80 + 92 + 88 + 96) / 5 = 88,2$.

Если реальное количество сбоев в июле составило 94, то прогноз сбоев на

август уже будет равен:
 $(80 + 92 + 88 + 96 + 94) / 5 = 90$
и так далее.

Число значений n для подсчета скользящей средней (в нашем примере $n=5$) выбирается в зависимости от того, насколько важны старые значения исследуемого показателя в сравнении с новыми. Так, если мы будем

использовать для подсчета трехмесячный период, тогда прогноз количества сбоев на июль составит:

$$(92 + 88 + 96)/3 = 92,3.$$

В случае с пятью месяцами старые значения имеют удельный вес $4/5$, а текущие $-1/5$. В случае с тремя месяцами, старые значения весят $2/3$, а текущие $-1/3$, т.е. скользящая средняя уже в большей степени зависит от текущего уровня и несколько слабее зависит от предшествующего.

Тренд–изменение, определяющее общее направление развития, основную тенденцию рядов динамики. Для выявления тренда ряда динамики можно использовать метод скользящей средней, в котором вместо фактического уровня берётся средняя, которая рассчитывается из нескольких уровней. Эта средняя будет скользящей, поскольку период усреднения постоянно меняется, вычитая один уровень и прибавляя другой.

$$Z_m = (z_m + z_{m+1} + \dots + z_{m+n-1})/n,$$

где n –количество выбранных значимых элементов временного ряда(выбранный период усреднения) для расчета скользящей средней, m –уровень расчета скользящей средней,

$$m = N - n, n, m < N, N - \text{длина временного ряда (количество его значений)}.$$

Скользящие средние не спрогнозируют изменения в тренде, а лишь просигналят об уже появившемся тренде.

Пример. Имеется следующий ряд динамики:

t	1	2	3	4	5	6	7	8	9	10	11	12
x	2	4	3	4	6	8	5	5	9	11	7	8

Определить значения всех скользящих средних, взяв период усреднения $n=3$. Построить в одной системе координат график временного ряда и график его тренда по полученным скользящим средним.

Вычислим значения скользящих средних.

$$Z_1 = (x_1 + x_2 + x_3) / 3 = (2 + 4 + 3) / 3 = 3 \text{ – первая скользящая средняя;}$$

$$Z_2 = (x_2 + x_3 + x_4) / 3 = (4 + 3 + 4) / 3 = 3,7 \text{ – вторая скользящая средняя;}$$

$$Z_3 = (x_3 + x_4 + x_5) / 3 = (3 + 4 + 6) / 3 = 4,3;$$

$$Z_4 = (x_4 + x_5 + x_6) / 3 = (4 + 6 + 8) / 3 = 6;$$

$$Z_5 = (x_5 + x_6 + x_7) / 3 = (6 + 8 + 5) / 3 = 6,5;$$

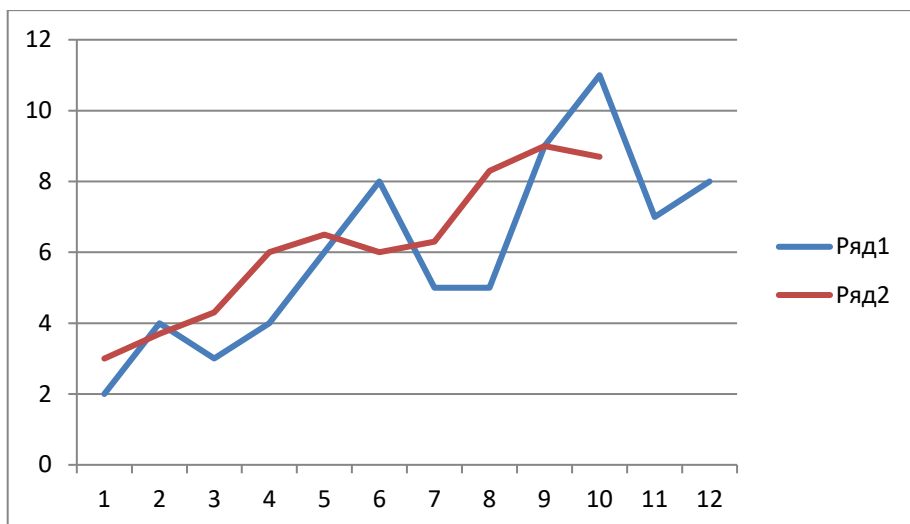
$$Z_6 = (x_6 + x_7 + x_8) / 3 = (8 + 5 + 5) / 3 = 6;$$

$$Z_7 = (x_7 + x_8 + x_9) / 3 = (5 + 5 + 9) / 3 = 6,3;$$

$$Z_8 = (x_8 + x_9 + x_{10}) / 3 = (5 + 9 + 11) / 3 = 8,3;$$

$$Z_9 = (x_9 + x_{10} + x_{11}) / 3 = (9 + 11 + 7) / 3 = 9 \text{ – девятая скользящая средняя;}$$

$$Z_{10} = (x_{10} + x_{11} + x_{12}) / 3 = (11 + 7 + 8) / 3 = 8,7 \text{ – десятая скользящая средняя.}$$



Лабораторная работа.

Даны временные ряды. Сделать расчет для двух значений периода усреднения $n_1=4$ и $n_2=5$. Сравнить результаты.

1.

t	1	2	3	4	5	6	7	8	9	10	11
Объем операций, млрд руб.	75	60	65	40	43	28	20	22	16	20	18

2.

t	1	2	3	4	5	6	7	8	9	10	11
Объем ВВП, млрд долл.	5,0	5,3	6,2	7,0	7,5	8,2	9,0	9,5	10,3	11,0	11,2

3.

t	1	2	3	4	5	6	7	8	9	10	11
Выручка, млн руб.	95	63	50	32	15	20	28	53	70	90	95

4.

t	1	2	3	4	5	6	7	8	9	10	11
Объем операций, млн руб.	5	25	40	50	55	60	61	62	63	64	65

5.

t	1	2	3	4	5	6	7	8	9	10	11
Цена, долл/барр	16,1	18,52	18,23	12,21	17,25	26,2	22,81	23,74	26,78	33,64	34,05

6.

t	1	2	3	4	5	6	7	8	9	10	11
Экспорт	171,6	164,07	165,74	186,17	185,67	184,16	186,18	186,42	204,4	232,76	233,54

7.

t	1	2	3	4	5	6	7	8	9	10	11
Объем рынка, млн.долл.	15,1	17,53	18,43	11,21	16,25	25,2	21,81	24,74	27,78	33,64	34,05

8.

t	1	2	3	4	5	6	7	8	9	10	11
Объем продаж, млн.руб.	95	95	85	11,21	75	70	85	83	82	83	85

9.

t	1	2	3	4	5	6	7	8	9	10	11
Объем рынка, млн.долл.	17,92	20,67	19,99	19,72	23,97	28,50	24,44	25,02	28,83	27,27	29,05

10.

t	1	2	3	4	5	6	7	8	9	10	11
Объем продаж, млн.руб.	80	79	85	80	75	68	65	68	65	62	61

11.

t	1	2	3	4	5	6	7	8	9	10	11
Объем экспорта, млн тонн	129,7	116,0	115,8	116,8	127,2	124,6	132,5	162,1	174,3	208,5	201,3

12.

t	1	2	3	4	5	6	7	8	9	10	11
Цена, долл/барр	26,1	28,52	28,23	22,21	27,25	36,2	32,81	33,74	36,78	43,64	44,05

3.3 Метод наименьших квадратов

Пусть на отрезке $[a;b]$ задано множество точек $\omega_n = \{a = x_0 < x_1 < \dots < x_n = b\}$ и известны значения сеточной функции $\tilde{y}_i = \tilde{y}(x_i)$, $i = \overline{0, n}$, которые получены с погрешностью ε_i относительно некоторой функции $y(x)$, т.е. $\tilde{y}_i = y(x_i) \pm \varepsilon_i, i = \overline{0, n}$. Узлы сетки также могут быть заданы с некоторой погрешностью. Задача состоит в том, чтобы найти такую зависимость $f(x)$, которая наилучшим образом приближает функцию $y(x_i), i = \overline{0, n}$ по таблично заданной функции $\tilde{y}_i = y(x_i) \pm \varepsilon_i, i = \overline{0, n}$.

Практически вид приближающей функции $f(x)$ устанавливают следующим образом: по таблице строится точечный график функции $\tilde{y}_i = \tilde{y}(x_i)$, $i = \overline{0, n}$, а затем проводится плавная кривая, по возможности наилучшим образом отражающая характер расположения точек.

В качестве приближающих функций в зависимости от характера точечного графика функции $y(x)$ часто используют следующие функции:

- | | |
|---|---|
| 1. $f(x) = ax + b$;
(линейная) | 3. $f(x) = ax^m$;
(степенная) |
| 2. $f(x) = ax^2 + bx + c$;
(квадратичная) | 4. $f(x) = ae^{mx}$;
(экспоненциальная) |

5. $f(x) = \frac{1}{ax+b}$;
(дробно-линейная)

7. $f(x) = a\frac{1}{x} + b$;
(гиперболическая)

6. $f(x) = a \cdot \ln x + b$;
(логарифмическая)

8. $f(x) = \frac{x}{ax+b}$.
(дробно-рациональная)

Здесь a, b, c, m – параметры, подлежащие определению. Когда вид приближающей функции установлен, задача сводится только к отысканию этих параметров.

Будем предполагать, что выбор функциональной зависимости уже сделан, т.е. эмпирическую формулу можно записать в виде

$$f = f(x, c_0, c_1, \dots, c_m) \tag{3.1}$$

Необходимо найти оптимальное значение параметров c_0, c_1, \dots, c_m , которые будут минимизировать норму отклонения.

$$r(x_i) = r_i = f(x_i, c_0, \dots, c_m) - y(x_i), i = 0, 1, \dots, n \tag{3.2}$$

Для метода наименьших квадратов в качестве нормы берут сумму квадратов отклонений по всем узлам.

$$\|r\| = \sum_{i=0}^n (r_i)^2 = \sum_{i=0}^n (f(x_i, c_0, \dots, c_m) - y(x_i))^2 \tag{3.3}$$

Полагая, что норма отклонения является некоторой функцией параметров c_0, c_1, \dots, c_m , т.е.

$$\|r\| = S(c_0, c_1, \dots, c_m), \tag{3.4}$$

будем искать значения этих параметры c_0, c_1, \dots, c_m из условия минимизации функции $S(c_0, c_1, \dots, c_m)$. Это условие запишем следующим образом

$$\frac{\partial S}{\partial c_0} = 0, \frac{\partial S}{\partial c_1} = 0, \dots, \frac{\partial S}{\partial c_m} = 0 \tag{3.5}$$

Выписывая частные производные (3.5) в явном виде на основе нормы (3.3), получим соотношения, которые представляют собой $m+1$ уравнение относительно неизвестных c_0, c_1, \dots, c_m .

$$\begin{aligned} \frac{\partial S}{\partial c_0} &= 2 \sum_{i=0}^n [f(x_i, c_0, c_1, \dots, c_m) - y_i] f'_{c_0}(x_i, c_0, \dots, c_m) = 0 \\ \frac{\partial S}{\partial c_1} &= 2 \sum_{i=0}^n [f(x_i, c_0, c_1, \dots, c_m) - y_i] f'_{c_1}(x_i, c_0, \dots, c_m) = 0 \\ &\dots \dots \dots \tag{3.6} \\ \frac{\partial S}{\partial c_m} &= 2 \sum_{i=0}^n [f(x_i, c_0, c_1, \dots, c_m) - y_i] f'_{c_m}(x_i, c_0, \dots, c_m) = 0 \end{aligned}$$

Решая систему (3.6), найдем значения параметров c_0, c_1, \dots, c_m , которые в конечном итоге определяют вид функции $f = f(x, c_0, c_1, \dots, c_m)$.

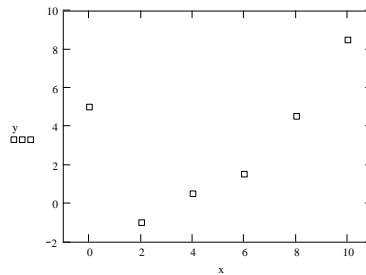
Пример. По заданной таблице значений функции построить точечный график и методом наименьших квадратов найти несколько приближающих аналитических функций. Сравнить качество полученных приближений. Совместить в одной плоскости графики исходной и найденных функций.

Имеем таблицу

x	0	2	4	6	8	10
y	5	-1	0.5	1.5	4.5	8.5

$n=6$

Строим точечный график



Полагаем, что x и y связаны линейной зависимостью $f_1(x) = ax + b$.

Находим частные производные: $\frac{\partial f_1}{\partial a} = x$, $\frac{\partial f_1}{\partial b} = 1$ и составляем систему

$$\begin{cases} \sum_{i=1}^n (y_i - ax_i - b) \cdot x_i = 0 \\ \sum_{i=1}^n (y_i - ax_i - b) \cdot 1 = 0 \end{cases} \quad \text{Преобразуем систему к виду:} \quad \begin{cases} a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i = \sum_{i=1}^n y_i x_i \\ a \sum_{i=1}^n x_i + nb = \sum_{i=1}^n y_i \end{cases}$$

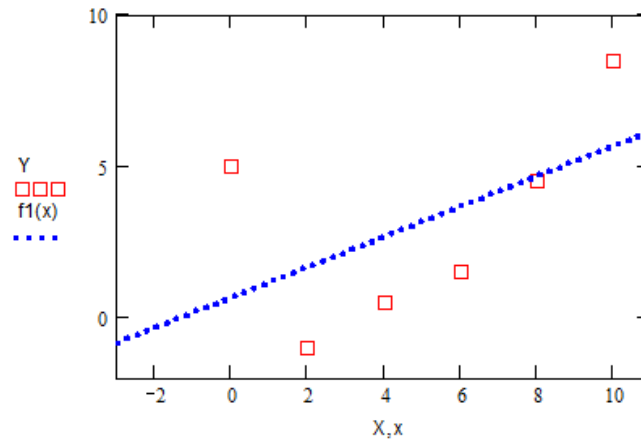
Для поиска параметров a и b построим таблицу следующего вида:

i	x_i	y_i	x_i^2	$x_i y_i$
1	0	5	0	0
2	2	-1	4	-2
3	4	0.5	16	2
4	6	1.5	36	9
5	8	4.5	64	36
6	10	8.5	100	85
Σ	30	19	220	130

По данным таблицы сформируем систему уравнений:

$$\begin{cases} 220 \cdot a + 30 \cdot b = 130 \\ 30 \cdot a + 6 \cdot b = 19 \end{cases}$$

Решив систему, получим $a = 0.5$ $b = 0.667$, тогда $f1(x) = 0.5 \cdot x + 0.667$.
 Строим график приближения



Пусть приближающая функция имеет вид квадратной регрессии (параболическая функция): $f2(x) = ax^2 + bx + c$

Находим частные производные: $\frac{\partial f2}{\partial a} = x^2$, $\frac{\partial f2}{\partial b} = x$, $\frac{\partial f2}{\partial c} = 1$ и состав-
 ляем систему

$$\begin{cases} \sum_{i=1}^n (y_i - ax_i^2 - bx_i - c) \cdot x_i^2 = 0 \\ \sum_{i=1}^n (y_i - ax_i^2 - bx_i - c) \cdot x_i = 0 \\ \sum_{i=1}^n (y_i - ax_i^2 - bx_i - c) \cdot 1 = 0 \end{cases}$$

Преобразуем систему к виду:

$$\begin{cases} a \sum_{i=1}^n x_i^4 + b \sum_{i=1}^n x_i^3 + c \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i x_i^2 \\ a \sum_{i=1}^n x_i^3 + b \sum_{i=1}^n x_i^2 + c \sum_{i=1}^n x_i = \sum_{i=1}^n y_i x_i \\ a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i + nc = \sum_{i=1}^n y_i \end{cases}$$

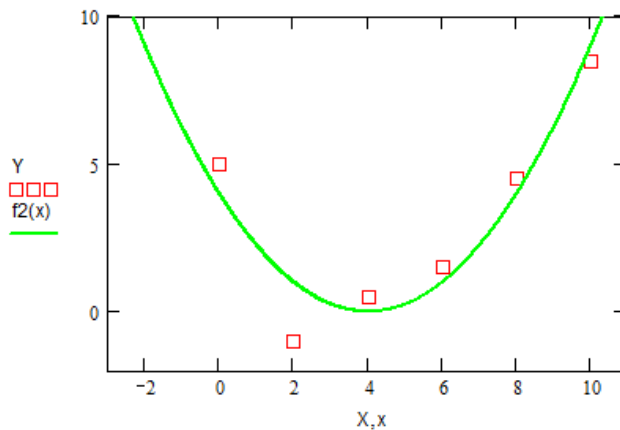
Для поиска параметров a , b построим таблицу следующего вида:

i	x_i	y_i	x_i^2	x_i^3	x_i^4	$x_i y_i$	$x_i^2 y_i$
1	0	5	0	0	0	0	0
2	2	-1	4	8	16	-2	-4
3	4	0.5	16	64	256	2	8
4	6	1.5	36	216	1296	9	54
5	8	4.5	64	512	4096	36	288
6	10	8.5	100	1000	10000	85	850
Σ	30	19	220	1800	15664	130	1196

По данным таблицы сформируем систему уравнений

$$\begin{cases} 15664a + 1800b + 220c = 1196 \\ 1800a + 220b + 30c = 130 \\ 220a + 30b + 6c = 19 \end{cases}$$

Решив систему, получим $a = 0.25$; $b = -2$; $c = 4$ и приближение $f_2(x) = 0.25x^2 - 2x + 4$. Отображаем графически



Пусть приближающая функция имеет вид кубической функции:
 $f_3(x) = ax^3 + bx^2 + cx + d$

Находим частные производные: $\frac{\partial f_3}{\partial a} = x^3$, $\frac{\partial f_3}{\partial b} = x^2$, $\frac{\partial f_3}{\partial c} = x$, $\frac{\partial f_3}{\partial d} = 1$ и

составляем систему.

$$\begin{cases} \sum_{i=1}^n (y_i - ax_i^3 - bx_i^2 - cx_i - d) \cdot x_i^3 = 0 \\ \sum_{i=1}^n (y_i - ax_i^3 - bx_i^2 - cx_i - d) \cdot x_i^2 = 0 \\ \sum_{i=1}^n (y_i - ax_i^3 - bx_i^2 - cx_i - d) \cdot x_i = 0 \\ \sum_{i=1}^n (y_i - ax_i^3 - bx_i^2 - cx_i - d) = 0 \end{cases}$$

Преобразуем систему к виду:

$$\begin{cases} a \sum_{i=1}^n x_i^6 + b \sum_{i=1}^n x_i^5 + c \sum_{i=1}^n x_i^4 + d \sum_{i=1}^n x_i^3 = \sum_{i=1}^n y_i x_i^3 \\ a \sum_{i=1}^n x_i^5 + b \sum_{i=1}^n x_i^4 + c \sum_{i=1}^n x_i^3 + d \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i x_i^2 \\ a \sum_{i=1}^n x_i^4 + b \sum_{i=1}^n x_i^3 + c \sum_{i=1}^n x_i^2 + d \sum_{i=1}^n x_i = \sum_{i=1}^n y_i x_i \\ a \sum_{i=1}^n x_i^3 + b \sum_{i=1}^n x_i^2 + c \sum_{i=1}^n x_i + nd = \sum_{i=1}^n y_i \end{cases}$$

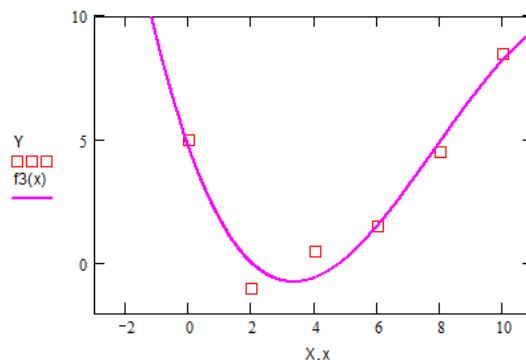
Для поиска параметров a , b построим таблицу следующего вида:

i	x_i	y_i	x_i^2	x_i^3	x_i^4	x_i^5	x_i^6	$x_i y_i$	$x_i^2 y_i$	$x_i^3 y_i$
1	0	5	0	0	0	0	0	0	0	0
2	2	-1	4	8	16	32	64	-2	-4	-8
3	4	0.5	16	64	256	1024	4096	2	8	32
4	6	1.5	36	216	1296	7776	46660	9	54	324
5	8	4.5	64	512	4096	32770	262100	36	288	2304
6	10	8.5	100	1000	10000	100000	1000000	85	850	8500
Σ	30	19	220	1800	15664	141602	1312920	130	1196	11152

По данным таблицы сформируем систему уравнений

$$\begin{cases} 1312920a + 141602b + 15664c + 1800d = 11152 \\ 141602a + 15664b + 1800c + 220d = 1196 \\ 15664a + 1800b + 220c + 19d = 130 \\ 1800a + 220b + 19c + 6d = 19 \end{cases}$$

Решив систему, получим $a = -0.029$; $b = 0.684$; $c = -3.586$; $d = 4.694$ и приближение $f_3(x) = -0.029x^3 + 0.684x^2 - 3.586x + 4.694$. Отображаем графически



Построив три функции приближения к исходной, находим для каждой сумму квадратов отклонений по формуле:

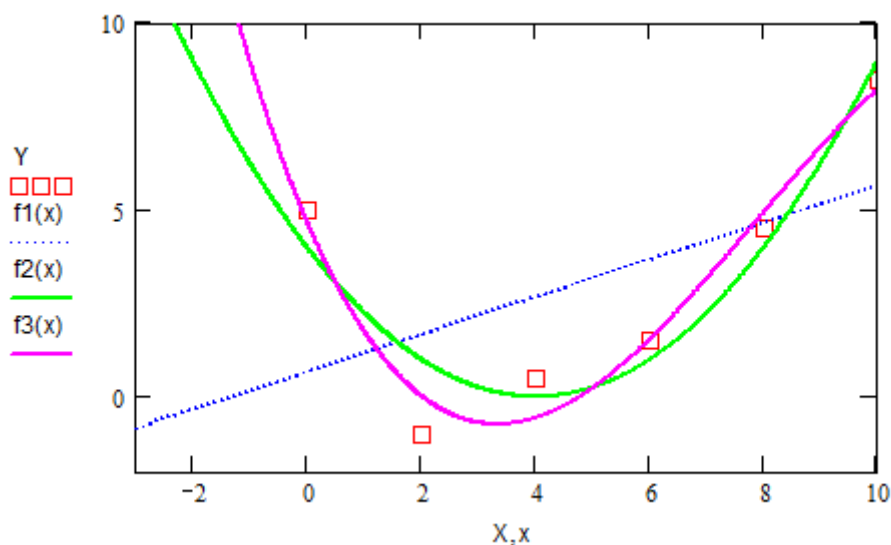
$$Q = \sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n (y_i - f(x_i))^2$$

Имеем таблицу:

i	x_i	y_i	$f_1(x)$	$f_2(x)$	$f_3(x)$	ε_1^2	ε_2^2	ε_3^2
1	0	5	0.667	4	4.694	18.775	1	0.094
2	2	-1	1.667	1	0.026	7.992	4	1.053
3	4	0.5	2.667	0	-0.562	6.185	0.25	1.128
4	6	1.5	3.667	1	1.538	7.007	0.25	0.0014
5	8	4.5	4.667	4	4.934	0.651	0.25	0.188
6	10	8.5	5.667	9	8.234	4.133	0.25	0.071
Q						44.743	6	2.5354

Из трех приближений выбираем то, для которого сумма квадратов отклонений минимальна. В нашем случае наилучшим приближением будет функция $f_3(x) = -0.029x^3 + 0.684x^2 - 3.586x + 4.694$.

Совместим в одной плоскости графики исходной и найденных функций:



Лабораторная работа

По заданной таблице значений функции построить точечный график и методом наименьших квадратов найти несколько приближающих аналитических функций. Сравнить качество полученных приближений. Совместить в одной плоскости графики исходной и найденной функций.

1.

x	-3	-2	-1	0	1	2	3
y	2,6	-0,3	-2	-2,3	-1,5	0,7	3,2

2.

x	0	2	4	6	8	10
y	5	-1	0,5	1,5	4,5	8,5

3.

x	1	2	3	4	5	6
y	1,14	2,78	4,07	4,91	5,41	5,52

4.

x	-3	-2	-1	0	1	2	3
y	0,71	-0,01	0,51	0,82	0,88	0,81	0,49

5.

x	-4	-3	-2	-1	0	1	2	3	4
y	6	3	1	0,3	-0,1	-0,2	0	0,2	1

6.

x	- 3	-2	-1	0	1	2
y	- 1,4	- 4,3	- 5,20	- 4,1	- 1,1	4,2

7.

x	7	8	9	10	11	12	13
y	3,1	4,9	5,3	5,8	6,1	6,1	5,9

8.

x	2	4	6	8	10
y	4,5	7,0	8,0	7,5	9,0

9.

x	0,30	0,91	1,50	2,00	2,20	2,62	3,00	3,30
y	0,20	0,43	0,35	0,52	0,81	0,68	1,15	0,85

10.

x	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7
y	3,02	2,81	2,57	2,39	2,18	1,99	1,81	1,85

11.

x	0	7	12	17	22	27	32	37
y	100	87,3	72,9	63,2	54,7	47,5	41,4	36,3

12.

x	0	0,5	1,0	1,5	2,0	2,5	3,0	3,5
y	1,67	1,32	1,10	0,81	0,48	0,18	- 0,10	- 0,46

ЛИТЕРАТУРА

1. Суровцев, И.С. Нейронные сети / И.С. Суровцев, В.И. Клюкин, Р.П. Пивоварова. – Воронеж: ВГУ, 1994. – 224 с.
2. Волошин, Г.Я. Статистические методы решения задач распознавания, основанные на аппроксимационном подходе / Г.Я. Волошин, И.А. Бурлаков, С.Т. Косенкова. – Владивосток: Изд-во Дальневост. отд-я РАН, 1992. – 523 с.
3. Созыкин, А.В. Обзор методов обучения глубоких нейронных сетей / А.В. Созыкин // Вестн. ЮУрГУ. – 2017. – Т. 6, № 3. – С. 28–59.
4. Афонский, А.А. Цифровые анализаторы спектра, сигналов и логики / А.А. Афонский, В.П. Дьяконов. – М.: СОЛОН-Пресс, 2009. – 248 с.
5. Сергиенко, А.Б. Цифровая обработка сигналов / А.Б. Сергиенко. – СПб.: Питер, 2006. – 751 с.

Учебное издание

КОРЧЕВСКАЯ Елена Алексеевна

ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

Методические рекомендации

Технический редактор

Г.В. Разбоева

Компьютерный дизайн

Л.В. Рудницкая

Подписано в печать 07.10. 2022. Формат 60x84 ¹/₁₆. Бумага офсетная.

Усл. печ. л. 2,56. Уч.-изд. л. 1,44. Тираж 25 экз. Заказ 177.

Издатель и полиграфическое исполнение – учреждение образования
«Витебский государственный университет имени П.М. Машерова».

Свидетельство о государственной регистрации в качестве издателя,
изготовителя, распространителя печатных изданий

№ 1/255 от 31.03.2014.

Отпечатано на ризографе учреждения образования
«Витебский государственный университет имени П.М. Машерова».

210038, г. Витебск, Московский проспект, 33.