

НЕЙРОСЕТЕВОЕ МОДЕЛИРОВАНИЕ ДИНАМИКИ НЕЛИНЕЙНЫХ СИСТЕМ

К.Ю. Гусев, В.Л. Бурковский

В статье рассматривается моделирование сложных нелинейных систем на основе аппарата нейронных сетей. Используется не классическая нейронная сеть, а нейронно-нечёткий модуль, объединяющий преимущества как нейронной сети, так и аппарата нечёткой логики. Также рассматриваются сервисные средства увеличения точности и скорости функционирования модуля. На уровне модели предлагаются модернизация архитектуры модуля и предобработка сигнала

Ключевые слова: нейронная сеть, нечёткая система, программный комплекс

Сегодня на первый план выходит нелинейная динамика, связанная с поиском единых механизмов в нелинейных системах различной природы. Модель объекта, которым может быть как отдельная отрасль, так и совокупность отраслей, позволяет анализировать множество потенциальных путей развития объекта (или отдельных характерных черт явления, сопровождающего развитие объекта) и выбрать наиболее рациональный путь его развития, в соответствии с целями и задачами, которые ставит исследователь. Для поиска таких путей развития в сложных нелинейных системах используются специальные математические методы.

Необходимость разработки новых методов прогнозирования обусловлена рядом причин. Основные методы, которые использовались ранее, при планировании и прогнозировании, можно разделить на три группы: балансовые, исследования операций, эконометрические методы. Их общей чертой можно назвать линейность (или сводимость к ней, посредством процедуры линеаризации соответствующих систем), которая является как недостатком, так и достоинством этих методов. Предположение о линейной зависимости является сильным упрощением действительности. Линейный характер связей принимается как постулат, подтвержденный достаточно обширным эмпирическим материалом.

К основным постулатам линейного мышления, которые следуют из детерминистских представлений о мире и математическом способе его описания с помощью систем линейных дифференциальных уравнений, относятся следующие [1]:

- возможность описания большинства процессов с большой степенью точности с помощью линейных уравнений или их комбинаций. Нелинейные члены представляют собой лишь небольшие добавки, не вносящие существенных качественных изменений в общую картину;

- однозначность стационарного решения в системе линейных уравнений. Это означает, что практически при любых условиях (параметрах системы) существует единственное стационарное решение (или не существует вовсе, что маловероятно), которое достигается независимо от начальных условий. В этом выражается представление о единственно верной цели, к которой следует стремиться любыми способами;

- устойчивость решения по отношению к виду уравнений и начальным условиям. Малые отклонения мало влияют на решения. Это соответствует представлениям об объективной закономерности, на которую фактически не могут повлиять личности и обстоятельства;

- возможность однозначной идентификации параметров в системе в случае полностью наблюдаемого вектора состояний (по совокупности фактических данных). Это означает, что последствиям можно однозначно определить причину

- к линейно-детерминистскому подходу следует отнести и представление о возможности выделения определяющего, лимитирующего фактора в любом процессе, о существовании ключевого управляющего фактора, обнаружив который, можно инициировать необходимый процесс.

Во всех сколько-нибудь сложных системах присутствуют свойства, которые могут быть описаны с помощью нелинейных моделей, и для них естественны: ограниченность решений, колебательные имульстационарные режимы, квазистохастическое пространственное и временное поведение. С учетом

свойств, присущих сложным системам, необходимо перейти к нелинейному принципу, который более адекватно подходит к моделированию современных технических, информационных и социально-экономических систем.

Математическая модель Априорный анализ данных

Априорный анализ является важным шагом при применении обучаемых нейросетей [2] и определяет скорость обучения, величины ошибок обучения и обобщения и иные свойства сети. Для предобработки количественных величин чаще всего применяют линейный сдвиг интервала значения признака, например, в интервал $[-1,1]$. Формула пересчета значения признака x для i -го примера выборки в интервал $[a,b]$ такова:

$$\tilde{x}_i = \frac{(x_i - x_{\min})(b - a)}{(x_{\max} - x_{\min})} + a.$$

При отсутствии жестких ограничений на диапазон значений предобработанного признака может быть выполнено масштабирование, дающее нулевое среднее и единичную дисперсию предобработанной величине, по формуле:

$$\tilde{x}_i = \frac{x - M(x)}{\sigma}.$$

где $M(x)$, $\sigma(x)$ - исходное выборочное среднее и среднее квадратичное отклонение соответственно.

Получение нулевых средних для входных сигналов сети ускоряет градиентное обучение, поскольку снижает отношение максимального и минимального ненулевого собственных чисел матрицы вторых производных целевой функции по параметрам сети [3].

Иногда проводят и предварительную (перед линейным масштабированием) нелинейную предобработку - например, логарифмирование. При одновременном же рассмотрении всего набора независимых признаков можно убрать линейные корреляции между признаками, что также положительно влияет на скорость обучения [4].

Предлагается использовать два показателя для анализа входных данных - коэффициент Херста и константы Липшица. Эти показатели служат для индикации готовности данных к прогнозированию.

Для вычисления коэффициента Херста применяется метод анализа временных рядов, позволяющий определить наличие непериодических циклов и памяти ряда - RS-анализ.

Статистический RS-анализ есть весьма эффективный метод обнаружения не только эффектов «сильной памяти», но и для обнаружения периодических и непериодических циклов.

Рассмотрим подробно алгоритм анализа, состоящий из следующих шагов [5]:

Шаг 1. Формирование выборки состояния нелинейной системы за определённый промежуток времени.

Шаг 2. Вычисление логарифмической динамики ряда

$$h_i = \ln\left(\frac{x_i}{x_{i-1}}\right).$$

Шаг 3. Вычисление величины

$$H_n = \sum_{i=1}^n h_i.$$

Шаг 4. Вычисление математического ожидания и дисперсии

$$R_n = \max_{k=1,2,\dots,n} \left(H_k - \frac{k}{n} H_n\right) - \min_{k=1,2,\dots,n} \left(H_k - \frac{k}{n} H_n\right)$$

$$S_n^2 = \frac{1}{n} \sum_{k=1}^n h_k^2 - \left(\frac{1}{n} \sum_{k=1}^n h_k\right)^2.$$

Величина $\bar{h}_n = H_n/n$ есть эмпирическое среднее построенное по выборке (h_1, h_2, \dots, h_n) .

Шаг 5. Определение приведённого размаха накопленных сумм

$$Q_n = \frac{R_n}{S_n}.$$

Шаг 6. Зафиксировать зависимость Q_n от n . Результат RS-анализа есть вычисление показателя Херста (H) - угол наклона средней линии зависимости Q_n от n , который находится с помощью метода наименьших квадратов. В зависимости от значения показателя анализируемый ряд можно охарактеризовать:

- $H > 0.5$ - система с долгой памятью (персистентный);
- $H < 0.5$ - отсутствие долговременной памяти (антиперсистентный);
- $H \approx 0.5$ - независимый процесс.

Для оценки сложности обучающей выборки предлагается рассчитывать значение константы Липшица [6] обучающей выборки. Константа Липшица выборки

$$\{x_i, y_i\}, i = 1 \dots N$$

Равна

$$L = \max_{i \neq j} \frac{|y_i - y_j|}{|x_i - x_j|},$$

Где x и y вектора входных сигналов и требуемых выходных сигналов соответственно.

В [7] показано, что константа Липшица выборки влияет на процесс обучения и свойства обученной сети, и при предобработке выборки нужно минимизировать константу Липшица. Поскольку константа Липшица выборки определяется примерами с разными ответными и близкими входными частями, то целенаправленно предобрабатывать нужно входные признаки, имеющие различные значения для этих конфликтных примеров.

Например, для возможности «подтягивания» хвостов распределений в качестве масштабирующей нелинейной функции взята сигмоида

$$\tilde{x} = \frac{x - M}{c + |x - M|},$$

Где M – выборочное среднее $M(x)$, а c подбирается таким образом, чтобы увеличить расстояние по x между примерами, определяющими константу Липшица выборки.

Также для улучшения используемых показателей предлагается ввести явную фильтрацию входной выборки.

- Фурье фильтрация основана на преобразовании Фурье и содержит следующие шаги:

Шаг 1. Прямое преобразование Фурье;

Шаг 2. Анализ спектра сигнала по Фурье образу и выбор типа фильтра и его параметров;

Шаг 3. Генерация фильтра и его применение к Фурье образу;

Шаг 4. Обратное преобразование Фурье для получения отфильтрованных данных.

Преимуществом Фурье фильтрации является возможность: в реальном масштабе времени анализировать спектр сигнала, на базе этого анализа гибко подбирать параметры фильтра и наблюдать результат фильтрации с последующей коррекцией параметров фильтра, если в этом есть необходимость.

- использование скользящих средних для выявления основных тенденций или циклов временных рядов [8].

Нечётко-нейронный модуль

Структурно гибридная модель нечётко-нейронного модуля описывается выражением

$$y(t+1) = \frac{\sum_{k=1}^N y_m \prod_{i=1}^n \exp \left[- \left(\frac{u - \bar{x}_i^k}{s_i^k} \right)^2 \right]}{\sum_{k=1}^N \prod_{i=1}^n \exp \left[- \left(\frac{u - \bar{x}_i^k}{s_i^k} \right)^2 \right]},$$

или, представляется в качестве модели типа «чёрный ящик», реализующую функцию f

$$y(t+1) = f(u_j).$$

Для оценки точности функционирования модели прогнозирования вводятся параметры качества:

- погрешность на выходе модуля – разность между эталонным и выходным значением модуля, абсолютная величина:

$$e = d - y(t+1);$$

- среднеквадратическая ошибка (мера погрешности) – функция погрешности, возведённая в квадрат, дифференцируемая функция:

$$\varepsilon = \frac{1}{2}(d - y)^2;$$

- доля периодов процесса, в которых прогнозируемое изменение (увеличение или уменьшение) и действительное изменения состояния экономического показателя (тренда) совпадают, измеряется в процентах:

$$z = \frac{K_p}{K_e} * 100\%,$$

где K_p – количество временных периодов, в которых наблюдается совпадение изменений, K_e – общее количество временных периодов прогнозируемых показателей.

Введём обозначения элементов нейронной сети. Пусть на вход нейронной сети поступают дискретные значения, которые представим в виде вектора

$$u = (u_1, u_2, \dots, u_n)$$

или, если представить каждое входной значение в виде элемента дискретного временного ряда,

$$u = (y(t), y(t-2), \dots, y(t-n)),$$

где n – количество входов нейросетевой модели.

Архитектура нечётко-нейронного модуля

Представив нейронную сеть в виде графа (рис. 1), вершинами которого являются функциональные модули, отметим наличие множества внутренних связей. Связи между слоями L1 и L2 являются наиболее значимыми при обучении нейронной сети и процессе функционирования.

Формализуем внутренние связи нейронной сети в виде двумерного бинарной матрицы:

- «1» - связь присутствует;
- «0» - связь отсутствует.

Такое представление упрощает процесс модификации внутренних связей для оптимизации скорости и точности функционирования системы. Поскольку задача поиска оптимальной матрицы внутренних связей многоэкстремальная и многокритериальная, что осложняет поиск глобального экстремума среди множества локальных. Предлагается использовать метод перебора (равномерного поиска) и генетический алгоритм для решения поставленной задачи. Выбор метода равномерного поиска не случаен, поскольку, не смотря на потраченные в процессе оптимизации ресурсы, будет найден глобальный экстремум. Применение данного метода предложено для проверки функционирования генетического алгоритма на определённом этапе оптимизации.

Проектирование архитектуры нейронно-нечёткого модуля состоит из следующих шагов [9] (в соответствии с типовым циклом эволюции):

Шаг 1. Декодирование каждой особи текущей популяции для описания архитектуры модуля.

Шаг 2. Обучение каждого модуля с архитектурой, полученной на первом шаге, с помощью заранее заданного правила (некоторые его параметры могут адаптивно уточняться в процессе обучения). Обучение должно начинаться при различных случайно выбираемых начальных значениях весов и параметров правила обучения.

Шаг 3. Оценивание приспособленности каждой особи по достигнутым результатам обучения, то есть по наименьшей целой среднеквадратичной погрешности обучения либо на основе тестирования, если наибольший интерес вызывает способность к обобщению, наименьшая длительность обучения или упрощение архитектуры (минимизация количества нейронов и связей между ними).

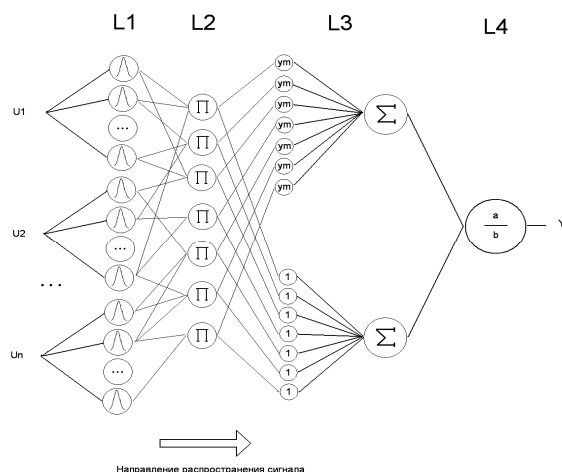


Рис. 1. Структура модуля

Шаг 4. Репродукция особей с вероятностью, соответствующей их приспособленности или рангу в зависимости от использования метода селекции.

Шаг 5. Формирование нового поколения в результате применения таких генетических операторов, как скрещивание, мутация и/или инверсия.

Нечёткая система прогнозирования (технический анализ)

Однако, временной ряд, составленный из ретроспективных данных конкретного интегрального экономического показателя, имеет составляющие, которые прогнозируются с помощью средств технического анализа. Для автоматизации процесса прогнозирования, использующего технический анализ, предлагается модель нечёткой системы с внешней базой знаний [10], где в качестве формы представления нечётких множеств используется функция Гаусса;

$$\mu(a_i) = \exp \left[- \left(\frac{a_i - \bar{x}_i^k}{s_i^k} \right)^2 \right]; i = 1 \dots n,$$

где n – количество входов нечёткой модели, а параметры функции имеют физическую интерпретацию: x – это центр, а s – ширина гауссовой кривой. Количество и значения параметров гауссовых функций подбираются таким образом, чтобы охватывать весь амплитудный диапазон обучающей выборки.

Первым этапом функционирования данной модели является составление из обучающей выборки базы знаний в виде пар «входной

вектор»-«прогнозируемое значение». Данные в базе знаний сохраняются не в абсолютных значениях, а в виде функций принадлежности к соответствующим гауссовым кривым [9].

На втором этапе, используя сформированную базу знаний, составляется прогноз, путём вычисления степени принадлежности имеющейся ситуации и эталонной. Нечёткая система даёт возможность вычисления результата несколькими способами, в частности:

- прямое нахождение ситуации в базе знаний, имеющей наибольшую степень принадлежности к входной. Результатом является совпавшее с найденной ситуацией «прогнозируемое значение»;

- вычисление результата по формуле

$$y = \frac{\sum y_i \cdot v_i}{\sum v_i}; i = 1 \dots n.$$

Программный комплекс

Программный комплекс прогнозирования состояния систем с нелинейной динамикой развития (forecast of non-linear systems - FNLS) состоит из двух функциональных модулей (нечёткий и нейронно-нечёткий модуль) и набора сервисных средств (реализация метода пробных шагов, фильтрации, модернизации архитектуры сети). Математической основой являются методы, описанные выше. В программном комплексе удачно совмещены как математические, так и эконометрические методы анализа и прогнозирования состояния систем. Выбор использования эконометрических методов и средств обусловлен тем, что в настоящее время прогнозирование социально-экономических систем является наиболее сложным и актуальным вопросом. Поэтому и методы, с успехом используемые для составления прогнозов для таких систем, являются наиболее эффективными. Так как современные технические, технологические и информационные системы не уступают по сложности описания и анализа социально-экономическим системам, то использование эконометрических методов повышает эффективность прогнозирования и подобных систем.

Русскоязычный интерфейс программного модуля (рис. 2) прост и интуитивно понятен для пользователя. Наличие нескольких вкладок для отображения результатов функционирования системы позволяет проводить разносторонний анализ получаемых результатов.

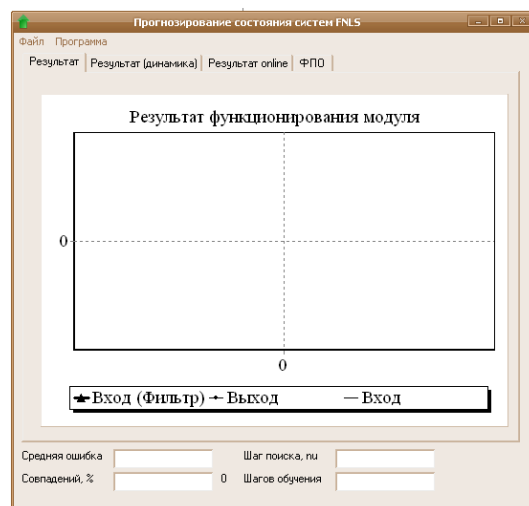


Рис. 2. Интерфейс программного комплекса

Основными режимами функционирования модуля являются режим «Обучения» и режим «Прогнозирования»

Функционирование online

Для апробации функционирования комплекса были использованы данные с мировых торговых площадок. Доступ к таким данным имеют специализированные информационно-торговые платформы, предназначенные для организации дилингового обслуживания на рынках. Для получения данных использовалась платформа MetaTrader 4 (общий интерфейс на рис. 3). В данной программе имеется собственная среда программирования MetaEditor для написания пользовательских индикаторов и советников. В программе имеется широкая база готовых индикаторов, советников и скриптов, но для поставленной программным комплексом задачи было принято решение написать пользовательский скрипт.

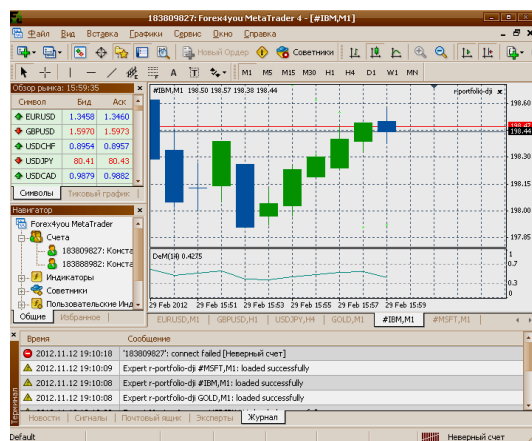


Рис. 3. MetaTrader 4

Скрипт должен циклично запрашивать данные с рынка и сохранять их в БД или в файл, доступные для программного комплекса прогнозирования. Программный комплекс, в свою очередь, как только распознаёт обновление файла или записи в БД автоматически запускает процесс обработки данных (например, если включена опция «Фильтрации») и прогнозирования на заранее заданный интервал времени.

Для подобной операции необходимо периодически обучать комплекс на новой выборке данных. Частота обучения зависит от интервала прогнозирования. Включение данной опции возможно в окне «Параметры» флагом «Таймер дискретизации online».

Результаты

Проведён всесторонний анализ программного комплекса и методов, применяемых при его моделировании, и получены результаты.

На рис. 4 и 5 – результаты обучения нечётко-нейронного модуля – средняя ошибка обучения модуля и данные, полученные после обучения модуля. При обучении модуля использовалась фильтрация входных данных. Коэффициент обучения (скорость обучения) был равен 0,7.



Рис. 4. Ошибка обучения

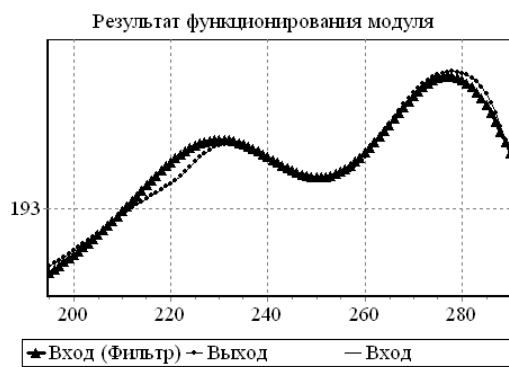


Рис. 5. Результат функционирования

Литература

1. Ризниченко Г.Ю. Нелинейное естественнонаучное мышление и экологическое сознание // Синергетическая парадигма; М.: Прогресс-Традиция, 2000, 256с.
2. Миркес Е.М; Нейрокомпьютер: проект стандарта, Новосибирск: Наука, 1999, 337с.
3. Горбань А.Н., Россиев Д.А. Нейронные сети на персональном компьютере; Новосибирск: Наука, 1996, 276с.
4. LeCun Y., Kanter I., Solla S.A. Second order properties of error surfaces: learning time and generalization / Advances in Neural Information Processing Systems 3; Morgan-Kaufmann, 1991, p. 924.
5. Э. Петерс. Фрактальный анализ финансовых рынков: применение теории хаоса в инвестициях и экономике. М.: Интернет-трейдинг, 2004, 304с.
6. Hamalainen J.J., Jarvima I. Input projection method for safe use of neural networks based on process data / Proc. IJCNN'1998, Anchorage, Alaska, USA, 1998, p. 239.
7. Царегородцев В.Г. Предобработка обучающей выборки, выборочная константа Липшица и свойства обученных нейронных сетей // Материалы X Всеросс. семинара «Нейроинформатика и ее приложения», Красноярск, 2002, 185с.
8. Макл Н. Кан, Технический анализ, ПИТЕР, 2003, 282 с.
9. Рутковская Д., Пилинский М., Рутковский Л., Нейронные сети, генетические алгоритмы и нечёткие системы. М.: Горячая линия, 2006. 452 с.
10. Бэстэнс Ж.-Э., В.-М. Ван Ден Берг, Д. Вуд; Нейронные сети и финансовые рынки; М.: ТВП, 1977, 236 с.

Воронежский государственный технический университет

NEURAL NETWORK MODELING OF NONLINEAR DYNAMICS

K.Yu. Gusev, V.L. Burkovskiy

The article deals with modeling of complex nonlinear systems based on neural network apparatus. Not used classical neural network and fuzzy neural module that combines the advantages of both neural network and fuzzy logic. Also considered a means to increase service accuracy and speed operation of the module. At the level of modernization of the architecture model proposed module

Key words: neural network, fuzzy system, software system