

ОБУЧЕНИЕ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ МЕТОДОМ ПСЕВДОРЕГУЛЯРИЗАЦИИ ДЛЯ МНОГОШАГОВОГО ПРОГНОЗИРОВАНИЯ НА ПРИМЕРЕ ХАОТИЧЕСКОГО ПРОЦЕССА МАККЕЯ-ГЛАССА

Артем Чернодуб

Abstract: Рассматривается задача многошагового прогнозирования нелинейных динамических процессов. Описываются существующие нейросетевые методы прогнозирования на основе использования многослойного персептрона с линией задержек и рекуррентного многослойного персептрона. Описывается метод обучения нейросетей методом расширенного фильтра Калмана с вычислением производных методом обратного распространения во времени. Предлагается метод псевдорегуляризации для уменьшения эффекта исчезновения градиентов, что приводит к повышению качества прогнозирования в многошаговом случае. Приводятся результаты численных экспериментов на примере прогнозирования хаотического процесса Маккея-Гласса.

Keywords: псевдорегуляризация, рекуррентный многослойный персептрон, обратное распространение во времени, расширенный фильтр Калмана

ACM Classification Keywords: I.2.6 Learning, Connectionism and neural nets

Введение

Прогнозирование динамических процессов является актуальной научной задачей, имеющей множество приложений в теории управления, экономике, медицине, физике и других областях. Нейросетевые методы хорошо себя зарекомендовали как средство моделирования динамических систем при неизвестной априори математической модели динамической системы [Хайкин, 2001, с. 120]. Нейронная сеть может быть обучена на известных примерах моделируемой последовательности и затем использоваться для прогнозирования на новых, ранее не виденных нейросетью примерах данных.

Существует два базовых метода для наделения нейронных сетей на основе многослойных персептронов свойствами, необходимыми для обработки динамических данных: добавление линий задержек на вход сети и добавление рекуррентных связей внутри сети. В первом случае, известном также как «метод временного окна» (Time-Window Neural Networks, Finite Impulse Response Neural Networks, Time-Lagged Neural Networks, Focused Multilayered Perceptrons) [Хайкин, 2001, с. 799], [Gers et. al., 2001], нейронная сеть получает на вход вместе с текущим входным сигналом задержанные во времени прошлые значения входных сигналов. Обучение сети выполняется с вычислением производных по известному методу обратного распространения ошибки (Backpropagation, BP) и применением градиентного метода оптимизации первого или второго порядка. Плюсами этого подхода являются его простота и технологичность: по нашим оценкам, сейчас более чем в 90% случаев для прогнозирования используется именно такая схема. С другой стороны, при таком подходе количество и порядок задержек временного окна должны быть установлены априори. Если эти величины не будут соответствовать порядку динамического процесса, нейросеть будет обучаться плохо или вовсе не обучится. Вторым минусом этого подхода является плохое качество обеспечиваемых с его помощью многошаговых прогнозов, когда в качестве входных данных прогноза на новый шаг используются собственные прогнозные данные, полученные для предыдущих шагов.

Вторым базовым подходом введения динамики в нейросети прямого распространения является добавление внутренних рекуррентных связей в скрытые, входные или выходные слои нейросети (Recurrent Multilayered Perceptrons, Simple Recurrent Neural Network, Elman's Neural Network) [Хайкин, 2001, с. 925], [Осовский, 2002, с. 210], [Cernansky and Benuskova, 2003]. Для учета влияния прошлых тактов на текущий результат вводится вычисление специальных динамических производных. Динамические производные принадлежат к одному из двух видов: «обратное распространение во времени» (Backpropagation Through Time, BPTT) [Хайкин, 2001, с. 943], [De Jesus and Hagan, 2007] или «рекуррентное обучение в реальном времени» (Real-Time Recurrent Learning, RTRL) [Хайкин, 2001, с. 949], [De Jesus and Hagan, 2007]. В качестве оптимизационного метода для настройки весов сети может использоваться, как и в первом случае, любой градиентный оптимизационный алгоритм. Такие рекуррентные сети по своей структуре более соответствуют моделируемым динамическим процессам, и поэтому лучше себя проявляют в задачах управления и многошагового прогнозирования [Prokhorov et. al., 2001]. В рекуррентном подходе нет необходимости априорного задания порядка линии задержек на входе, то есть, фактически, отсутствует необходимость предварительного выполнения структурной идентификации динамической системы. Вместе с тем, обучение таких сетей является более трудной задачей из-за дополнительных степеней свободы у таких сетей, на практике процесс обучения часто не сходится. Кроме того, при расчете динамических производных в перцептроно-подобных сетях имеет место эффект исчезновения градиента (gradient vanishing) [Хайкин, 2001, с. 968], [Hochreiter et. al, 2001], что дополнительно затрудняет для алгоритма обучения выявление корреляций между прошлыми входами и текущими целевыми выходами сети. Последний факт часто имеет решающее значение даже для процессов небольшого порядка, для борьбы с эффектом исчезновения градиента предлагаются новые архитектуры нейросетей на основе кратко-долговременной памяти (Long-Short Term Memory, LSTM) [Hochreiter and Schmidhuber, 1997] и гибридные с первым подходом сети, например – нейросети нелинейной авторегрессии с внешней моделью входов (Nonlinear Autoregression with eXternal inputs, NARX) [Хайкин, 2001, с. 936].

Мы предлагаем оригинальный подход для обучения рекуррентных нейросетей путем введения ограничений на величину градиента при обратном распространении ошибки во времени в целевую функцию оптимизации обучения нейросети, называемый нами псевдорегуляризацией. В качестве функции оптимизации используется модификация метода обучения расширенного фильтра Калмана, многокритериальный расширенный фильтр Калмана [Chernodub, 2012b]. Проводится экспериментальное сравнение предлагаемого метода обучения с обычным методом обучения BPTT(h) для рекуррентных многослойных перцептронов, а также многослойных перцептронов с линией задержек на примере многошагового прогнозирования хаотического процесса Маккея-Гласса [Alessandri et. al., 2001], [Bone and Cardot, 2011].

1. Многослойный перцептрон с линией задержек

Схема многослойного перцептрона с линией задержек порядка N представлена на рис. 1. Перцептрон содержит нейроны с линейной функцией активации для нейронов входного слоя и сигмоидальной функцией активации для нейронов скрытого и выходного слоев (в наших экспериментах мы использовали функции активации гиперболического тангенса). Весовые значения между слоями нейронов задаются матрицами $W^{(1)}$ и $W^{(2)}$. Перцептрон получает на вход $x(k)$ текущее значение временного ряда $y(k)$, а также ряд задержанных прошлых значений $y(k-1)$, $y(k-2)$, ..., $y(k-N)$, реализуемых с

помощью элементов запаздывания z^{-1} , z^{-2} , ..., z^{-N} , и по этим данным обучается делать прогноз следующего значения $\hat{y}(k+1)$.

Расчет выходного значения сети \tilde{y} выполняется по формуле:

$$\tilde{y} = g\left(\sum_j w_j^{(2)} f\left(\sum_i w_{ji}^{(1)} x_i\right)\right), \quad (1)$$

здесь: $w^{(1)}$ – веса нейронов скрытого слоя, $f(\cdot)$ – активационные функции нейронов скрытого слоя, $w^{(2)}$ – веса нейронов выходного слоя, $g(\cdot)$ – активационные функции нейронов выходного слоя. На рис. 2 показан нейроэмулятор с пятью нейронами в скрытом слое, который получает на вход состояние порядка N . Производные для обучения вычисляются обычным (статическим) методом обратного распространения ошибки. В нашей работе в качестве градиентного метода оптимизации мы использовали метод глобального расширенного фильтра Калмана (Global Extended Kalman Filter, GEKF)

[Cernansky and Benuskova, 2003], [Chernodub, 2012a], поэтому вместо градиентов $\frac{\partial [e(k)^2]}{\partial w}$ мы каждый

раз вычисляем якобианы $\frac{\partial \tilde{y}}{\partial w}$. Это делается путем пропуска значения 1 на каждом обратном проходе

вместо текущей ошибки обучения $e(k)$, что приводит к получению якобианов вместо градиентов при тех

же вычислениях, поскольку $\frac{\partial [e(k)^2]}{\partial w} = 2e(k) \frac{\partial y}{\partial w}$.

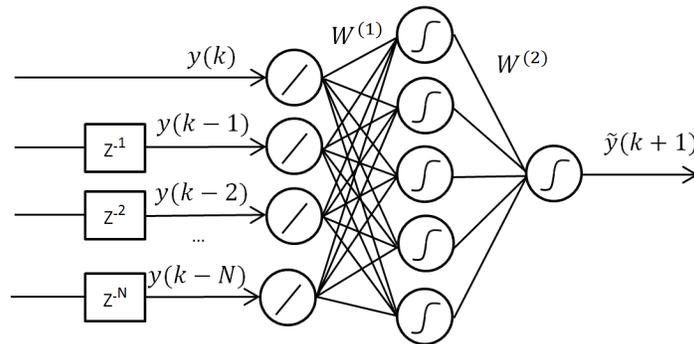


Рис. 1. Схема многослойного персептрона с линейными задержками на входе.

Для выполнения прогнозирования на H шагов вперед, проводится следующая итеративная процедура:

1. Персептрон выполняет прогнозирование на один шаг вперед, получает оценку следующей величины временного ряда $\hat{y}(k+1)$.
2. На основе имеющихся реальных прошлых входов и полученной оценки формируется новый входной вектор для нейросети $x(k+1) = [\hat{y}(k+1) \quad y(k) \quad \dots \quad y(k-N+1)]$.
3. Прогнозируется новое значение $\hat{y}(k+2)$.

Шаги 2) - 3) выполняются, пока не будет получена требуемая оценка $\hat{y}(k+H)$.

2. Рекуррентный многослойный персептрон

Рекуррентный многослойный персептрон (Recurrent Multilayer Perceptron, RMLP) (рис. 2) является модификацией многослойного персептрона, с добавлением обратных связей в скрытый слой нейронов, реализуемых задержками z^{-1} . Расчет выходного значения сети \tilde{y} аналогичен (1), различие состоит только в том, что в матрице весов скрытого слоя $W^{(1)}$ теперь также хранятся веса для рекуррентных связей.

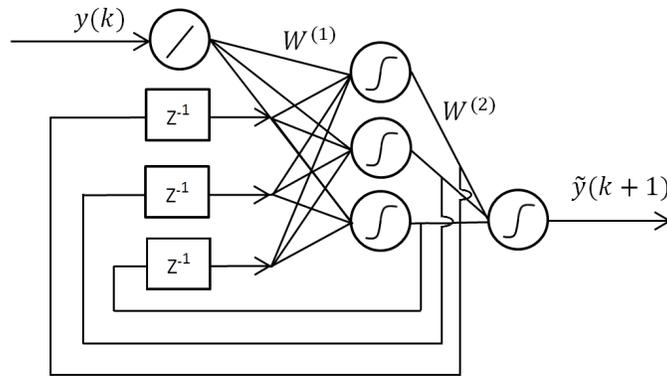


Рис. 2. Схема рекуррентного многослойного персептрона

Главным отличием в обучении рекуррентного многослойного персептрона от многослойного персептрона с линией задержек является использование динамических производных, рассчитываемых в нашем случае методом усеченного обратного распространения во времени ВРТТ(h) вместо статических производных, как для обучения многослойного персептрона с линией задержек. Схема вычисления производных во время обратного прохода представлена на рис. 3. На такте k , после вычисления выходного значения $\hat{y}(k+1)$, сеть разворачивается назад во времени на глубину усечения h .

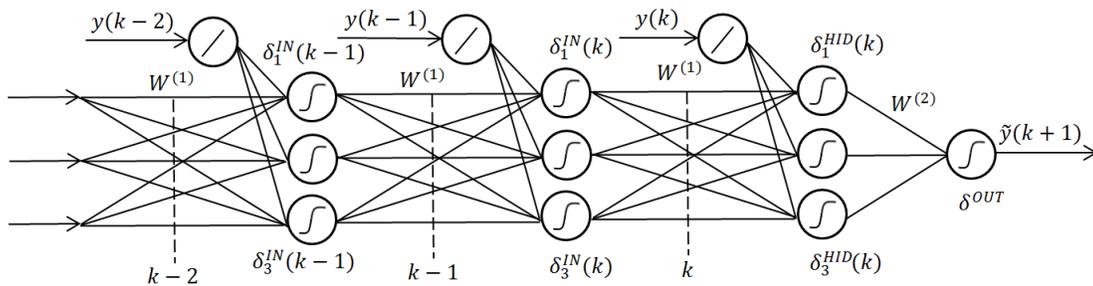


Рис. 3. Схема вычисления производных методом обратного распространения ошибки для рекуррентного многослойного персептрон, глубина усечения $h = 2$

При этом рекуррентная сеть представляется в виде многослойного персептрона прямого распространения с большим количеством слоев, где каждый слой соответствует одному из прошлых тактов $k-1, k-2, \dots, k-h$. Производные вычисляются методом обратного распространения, который применяется для полученной развернутой во времени сети прямого распространения. Локальные градиенты, вычисляемые в процессе процедуры обратного распространения, определяются следующими соотношениями:

$$\delta_j^{HD}(k) = f_j'(k)w_j^{(2)}\delta^{OUT}, \quad (2)$$

$$\delta_j^{IN}(k) = f_j'(k-1)\sum_{i=1}^K w_{ij}^{(1)}\delta_i^{HD}(k), \quad (3)$$

$$\delta_j^{IN}(k-n) = f_j'(k-n-1)\sum_{i=1}^K w_{ij}^{(1)}\delta_i^{IN}(k-n+1), \quad (4)$$

здесь: $w^{(1)}$ – веса нейронов скрытого слоя, $f_j(k)$ – активационная функция j -го нейрона скрытого слоя на такте k , $w^{(2)}$ – веса нейронов выходного слоя, δ_j^{HD} – локальный градиент для j -го нейрона скрытого слоя, $\delta_j^{IN}(k-n)$ – локальный градиент для j -го нейрона условного входного слоя на такте $k-n$, $1 \leq n \leq h$, h – глубина усечения.

3. Обучение нейросетей методом расширенного фильтра Калмана с псевдoreгуляризацией

При вычислении динамических производных методом ВРТТ(h) для персептроно-подобных нейросетей большую роль играет эффект исчезновения градиента, ухудшающий качество обучения нейросетей. Смысл этого эффекта можно интуитивно понять, взглянув на формулы вычисления локальных градиентов (2)-(4). Абсолютное значение каждого нового локальный градиент $\delta_j^{IN}(k-n)$ на практике получается меньше, чем предыдущего градиента $\delta_j^{IN}(k-n+1)$, поскольку является произведением функций по модулю меньших 1: пропускаемые в начале градиенты δ^{OUT} по определению всегда равны 1, веса нейронной сети $w_{ij}^{(1)}$ и $w_i^{(2)}$ не могут быть больше 1, поскольку это приводит к случаю переобучения, производные активационных функций $f(\cdot)$ всегда меньше 1. Более строгое доказательство эффекта исчезновения градиента и исследования на эту тему можно найти в [Hochreiter et. al, 2001] .

В настоящей работе мы предлагаем метод обучения нейросетей с оптимизацией локальных градиентов, названный нами псевдoreгуляризацией. Суть псевдoreгуляризации состоит в минимизации в процессе обучения нейроэмулятора функционала $Q(w)$:

$$Q(w) = e(w) + \lambda \left(\sum_{j=1}^K (1 - \delta_j^{IN}(w, k))^2 + \sum_{j=1}^K (1 - \delta_j^{HD}(w, k))^2 \right), \quad (5)$$

здесь: $e(w)$ – среднеквадратичная ошибка моделирования динамического процесса, δ_j^{HD} и δ_j^{IN} – локальные градиенты (2) и (3) соответственно, λ – заданная априори константа псевдoreгуляризации. Нашей целью является обучение нейросетей динамике моделируемого процесса с максимизацией абсолютных значений градиентов скрытых и входных нейронов в среднем (6),

$$\Omega = \frac{1}{N} \sum_{k=1}^N \sum_{j=1}^K (\delta_j^{IN}(k)^2 + \delta_j^{HD}(k)^2), \quad (6)$$

при этом учитывается, что в реальных системах вследствие эффекта исчезновения градиента $\delta(w) \ll 1$. В формуле (6) K – количество нейронов в скрытом слое, N – количество элементов обучающей выборки. На остальные локальные градиенты (4) ограничения аналогичные второму слагаемому (5) не накладываются, поскольку нам априори неизвестна степень влияния прошлых тактов работы системы на текущий такт. Тем не менее, это возможно – если это сделать, определение этих

ограничений будет по смыслу аналогично априорному определению временных задержек в многослойных сетях прямого распространения с линией задержек.

Для минимизации функционала (5) в настоящем работе мы предлагаем использовать обобщение метода оптимизации расширенного фильтра Калмана для многокритериального случая [Chernodub, 2012b]. Теперь выходом модели динамического процесса обучения на каждом такте является не только результат прямого прохода выход нейросети \tilde{y} , а и вычисляемые во время выполнения обратного прохода величины $\delta_j^{HID}(k)$ и $\delta_j^{IN}(k)$, $j=1,2,\dots,K$. Соответственно, для дополнительного выхода

рассчитываются якобианы $\frac{\partial[(\delta_j^{IN}(k))^2]}{\partial w}$ и $\frac{\partial[(\delta_j^{HID}(k))^2]}{\partial w}$ для включения в матрицу наблюдений H , а

второе слагаемое функционала $Q(w)$ добавляется в матрицу отклонений E .

Продифференцируем квадрат локального градиента (2) по весам скрытого и выходного слоя. Учитывая, что $\delta_j^{HID}(k)$ является функцией только от весов выходного слоя $w^{(2)}$:

$$\frac{\partial}{\partial w_{\alpha\beta}^{(1)}}(\delta_j^{HID}(k)^2) = 0, \quad (7)$$

$$\begin{cases} \frac{\partial}{\partial w_{\alpha}^{(2)}}(\delta_j^{HID}(k)^2) = 2\delta_j^{HID}(k)f_j'(k-1)w_{\alpha}^{(2)}\delta^{OUT}, & j = \alpha; \\ \frac{\partial}{\partial w_{\alpha}^{(2)}}(\delta_j^{HID}(k)^2) = 0, & j \neq \alpha; \end{cases} \quad (8)$$

Аналогично, продифференцируем квадрат локального градиента (3) по весам скрытого и выходного слоя:

$$\begin{cases} \frac{\partial}{\partial w_{\alpha\beta}^{(1)}}(\delta_j^{IN}(k)^2) = 2\delta_j^{IN}(k)f_j'(k-1)\delta_{\alpha}^{HID}, & j = \beta; \\ \frac{\partial}{\partial w_{\alpha\beta}^{(1)}}(\delta_j^{IN}(k)^2) = 0, & j \neq \beta; \end{cases} \quad (9)$$

$$\frac{\partial}{\partial w_{\alpha}^{(2)}}(\delta_j^{IN}(k)^2) = 2\delta_j^{IN}(k)f_j'(k-1)\sum_{i=1}^K w_{ij}^{(1)} \frac{\partial}{\partial w_{\alpha}^{(2)}}(\delta_i^{HID}(k)). \quad (10)$$

4. Обучение нейросетей методом расширенного фильтра Калмана с псевдорегуляризацией

Фильтр Калмана является эффективным рекурсивным фильтром, который по ряду зашумленных и неполных измерений позволяет оценить внутреннее состояние динамической системы и применяется в широчайшем спектре технических устройств, от спидометров автомобиля до радиоприемников и радаров. Обучение нейронных сетей является достаточно неожиданным применением для теории фильтрации Калмана [Хайкин, 2001, с. 960], [Prokhorov et. al., 2001], [Семанский, 2003] и, вместе с тем, очень эффективным: с одной стороны, качество такого обучения находится на уровне лучших пакетных алгоритмов второго порядка, таких как метод Левенберга-Марквардта или квази-Ньютоновских методов [Alessandri et. al., 2001], а с другой стороны – обучение проводится в режиме он-лайн, что является актуальным в случае выборок большого объема и задач управления. Обучение нейронной сети методом расширенного фильтра Калмана рассматривается как задача оценки истинного состояния некоторой неизвестной «идеальной» нейросети, обеспечивающей нулевое рассогласование, под состояниями в данном случае принимаются значения весов нейросети $w(k)$, а под рассогласованием – текущая ошибка обучения $e(k)$. Этот динамический процесс обучения может быть

описан парой уравнений в пространстве состояний, (11) и (12). Уравнение состояния (11) представляет собой модель процесса, представляющий собой эволюцию вектора весов под воздействием случайного процесса $\xi(k)$, который считается белым шумом с нулевым математическим ожиданием и известной диагональной ковариационной матрицей Q :

$$w(k+1) = w(k) + \xi(k). \quad (11)$$

Уравнение выхода (12) представляет собой линеаризованную модель нейросети (1) на такте k , зашумленную случайным процессом $\zeta(k)$, который считается белым шумом с нулевым математическим ожиданием и известной диагональной ковариационной матрицей R :

$$h(k) = \frac{\partial y(w(k), v(k), x(k))}{\partial w} + \zeta(k), \quad (12)$$

здесь: $w(k)$ – веса нейросети, $v(k)$ – постсинаптические потенциалы нейронов, $x(k)$ – входные значения сети. Вычисление мгновенных значений производных $\frac{\partial y}{\partial w}$ производится методом обратного распространения в случае многослойного персептрона с линией задержек или методом обратного распространения во времени в случае рекуррентного персептрона. Рассогласование $e(k)$ рассчитывается по формуле:

$$e(k) = t(k) - \tilde{y}(k), \quad (13)$$

здесь: $t(k)$ – целевое значение для нейросети, $\tilde{y}(k)$ – реальный выход нейросети, рассчитываемый по формуле (1).

Перед обучением нейросети проходит этап инициализации. Задаются ковариационные матрицы шума измерения $R = \eta I$ и динамического шума обучения $Q = \mu I$, размер матрицы R в нашем случае равен $(L + M \times K) \times (L + M \times K)$, размер матрицы Q равен $N \times N$, здесь: L – количество выходных нейронов, N – количество весовых коэффициентов нейросети, M – количество слоев, для которых оптимизируются локальные градиенты, K – количество нейронов скрытого слоя. У нас $L = 1$ и $M = 2$. Размер матрицы наблюдений H равен $(L + M \times K) \times N$, размер матрицы усиления Калмана \hat{K} равен $N \times (L + M \times K)$, размер матрицы корреляции процесса обучения P равен $N \times N$. Коэффициент η имеет смысл скорости обучения, у нас $\eta = 0.001$, коэффициент μ определяет шум измерения, у нас $\mu = 10^{-4}$.

На такте k выполняются следующие действия:

- 1) Выполняется «прямой проход» нейросети, вычисляется выход нейросети $\tilde{y}(k)$ согласно формуле (1).
- 2) Выполняется «обратный проход» нейросети: методом обратного распространения вычисляются якобианы $\frac{\partial \tilde{y}}{\partial w}$. В процессе расчета якобианов, параллельно вычисляются локальные градиенты для скрытого и входного слоя нейронов (2)-(4), вычисляются якобианы (7)-(10). Формируется многокритериальная матрица наблюдений $H(k)$:

$$H(k) = \begin{bmatrix} \frac{\partial \bar{y}}{\partial w_1} & \frac{\partial \bar{y}}{\partial w_2} & \dots & \frac{\partial \bar{y}}{\partial w_N} \\ \lambda \frac{\partial [(\delta_1^{HID})^2]}{\partial w_1} & \lambda \frac{\partial [(\delta_1^{HID})^2]}{\partial w_2} & \dots & \lambda \frac{\partial [(\delta_1^{HID})^2]}{\partial w_N} \\ \dots & \dots & \dots & \dots \\ \lambda \frac{\partial [(\delta_K^{IN})^2]}{\partial w_1} & \lambda \frac{\partial [(\delta_K^{IN})^2]}{\partial w_2} & \dots & \lambda \frac{\partial [(\delta_K^{IN})^2]}{\partial w_N} \end{bmatrix}^T. \quad (12)$$

- 3) Формируется многокритериальная матрица отклонений $E(k)$ размера $1 \times (L + M \times K)$ (здесь K – количество нейронов скрытого слоя):

$$\tilde{E}(k) = [e(k) \quad \lambda(1 - (\delta_1^{HID})^2) \quad \dots \quad \lambda(1 - (\delta_K^{HID})^2) \quad \lambda(1 - (\delta_1^{IN})^2) \quad \dots \quad \lambda(1 - (\delta_K^{IN})^2)] \quad (13)$$

- 4) Вычисляются новые значения весов нейросети $w(k+1)$ и матрицы корреляции $P(k+1)$ по формулам (14)-(16):

$$\hat{K}(k) = P(k)H(k)^T [H(k)P(k)H(k)^T + R]^{-1}, \quad (14)$$

$$P(k+1) = P(k) - \hat{K}(k)H(k)P(k) + Q, \quad (15)$$

$$w(k+1) = w(k) + \hat{K}(k)e(k), \quad (16)$$

Действия 1) - 4) выполняются для всех элементов обучающей выборки.

5. Задача прогнозирования процесса Маккея-Гласса

Для проведения экспериментов был использован стандартный для данного типа задач процесс Маккея-Гласса. Последовательность Маккея-Гласса задается разностным уравнением:

$$x_{t+1} = (1-b)x_t + a \frac{x_{t-\tau}}{1 + (x_{t-\tau})^{10}}, t = \tau, \tau + 1, \dots,$$

где $\tau \geq 1$ — целые числа. Для проведения экспериментов была сгенерирована последовательность из 3000 значений с использованием следующих значений параметров: $a = 0.2$, $b = 0.1$, $\tau = 17$, как в [Bone and Cardot, 2011].

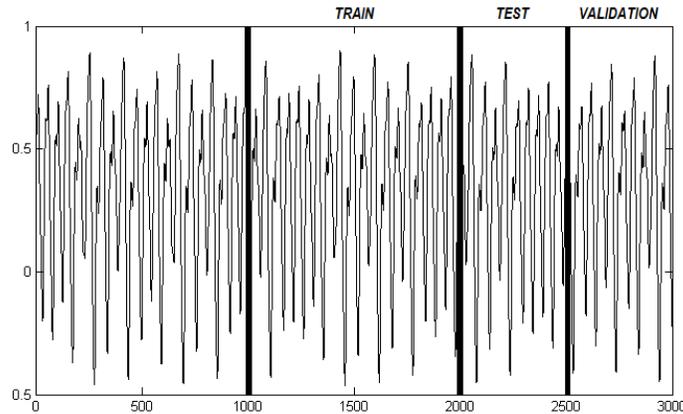


Рис. 4. Деление процесса Маккея-Гласса на обучающую (TRAIN), тестовую (TEST) и валидационную (VALIDATION) последовательности.

Первые 1000 значений сгенерированной последовательности были проигнорированы, 1000 следующих значений были использованы в качестве обучающей выборки, последующие 500 – в качестве тестовой выборки, оставшиеся 500 – в качестве валидационной выборки (рис. 4).

Качество многошагового прогнозирования оценивалось по критерию нормированной среднеквадратичной ошибки с горизонтом $NRMSE(H)$, определенной по формуле:

$$NRMSE(H) = \frac{1}{H} \sum_{k=1}^H \frac{\langle (y_k - t_k)^2 \rangle^{\frac{1}{2}}}{\langle (t_k - \langle t_k \rangle)^2 \rangle^{\frac{1}{2}}}, \quad (17)$$

где: \hat{y}_k — прогнозируемые значения, t_k — целевые значения, H — горизонт прогнозирования.

6. Экспериментальные результаты

Первоначально, для решения задачи прогнозирования было обучено 100 многослойных персептронов с линией задержек (MLP). Задача обучения для MLP ставилась фактически следующим образом: по N известным прошлым значениям последовательности $(y_k, y_{k-1}, \dots, y_{k-N})$ дать оценку \hat{y}_{k+1} следующего значения y_{k+1} . Количество нейронов в скрытом слое варьировалось от 3 до 8, обучение велось методом глобального расширенного фильтра Калмана. В процессе тренировки сети обучались на протяжении 100 эпох на обучающей последовательности и после каждой эпохи проверялись на способность к обобщению на тестовой последовательности. Сеть, показавшая лучший результат на тестовой последовательности, тестировалась на валидационной выборке. Этот результат считается окончательным и представлен в таблицах и на графиках.

Затем было обучено по 100 рекуррентных многослойных персептронов с вычислением динамических производных методом обратного распространения во времени (RMLP BPTT), были взяты глубины усечения $h=2$ и $h=5$. Количество нейронов в скрытом слое для этих сетей колебалось от 3 до 7, обучение велось также методом глобального расширенного фильтра Калмана на протяжении 50 эпох.

Наконец, было обучено по 100 рекуррентных сетей на протяжении 100 эпох с использованием псевдорегуляризации и мультикритериального расширенного фильтра Калмана для различных параметров величин псевдорегуляризации $\lambda=0.05$ и $\lambda=0.1$ и различных величин глубины усечения $h=2$ и $h=5$ (RMLP BPTT Pseudoregularization). Методы отбора сетей и тестирования аналогичны. Результаты представлены в табл. 1 и на рис. 5.

Таблица 1. Результаты многошагового прогнозирования процесса Маккея-Гласса

Название подхода	1	2	3	4	5	10	15
MLP	0.000623	0.012491	0.038825	0.079069	0.13231	0.54931	1.1021
RMLP BPTT, $h=2$	0.000303	0.00198	0.007092	0.018627	0.04029	0.36044	0.94303
RMLP BPTT, $h=5$	0.000667	0.003813	0.013268	0.034856	0.073757	0.48109	0.96754
RMLP BPTT PseudoR. $h=2, \lambda=0.05$	0.001611	0.008454	0.022531	0.045074	0.076353	0.32683	0.63858
RMLP BPTT PseudoR. $h=5, \lambda=0.05$	0.000944	0.003519	0.009875	0.021813	0.040582	0.22183	0.48481
RMLP BPTT PseudoR. $h=2, \lambda=0.1$	0.0012346	0.0064628	0.017244	0.034703	0.059456	0.27751	0.57487
RMLP BPTT PseudoR. $h=5, \lambda=0.1$	0.000435	0.001903	0.005786	0.013177	0.024796	0.1513	0.37158

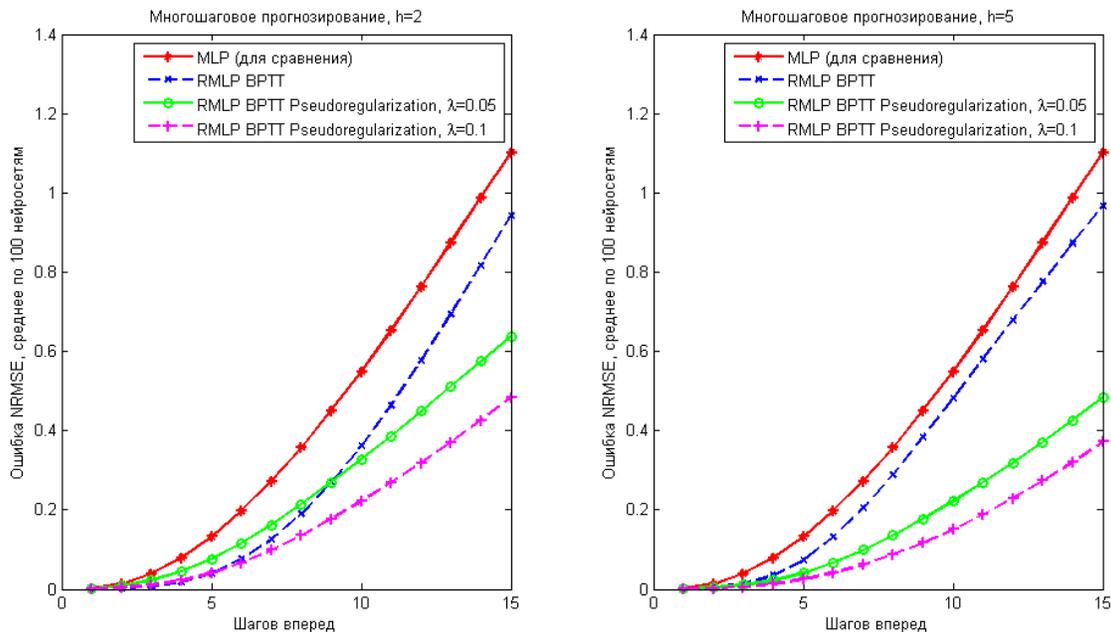


Рис. 5. Зависимость ошибки многошагового прогнозирования от горизонта прогнозирования, усреднение по 100 нейросетям

Выводы

В работе предложен новый метод обучения рекуррентных нейросетей с применением псевдорегуляризации и мультикритериального фильтра Калмана для уменьшения эффекта исчезновения градиента. Было проведено экспериментальное исследование предлагаемого метода, показавшее эффективность нового метода для многошагового прогнозирования по сравнению с традиционным обучением рекуррентных нейросетей методом расширенного фильтра Калмана. Использование псевдорегуляризации обеспечило уменьшение ошибки прогнозирования в среднем до 1.5 – 2 раз.

Благодарности

Работа опубликована при финансовой поддержке проекта **ITHEA XXI** Института информационных теорий и приложений FOI ITHEA Болгария www.ithea.org и Ассоциации создателей и пользователей интеллектуальных систем ADUIS Украина www.aduis.com.ua.

Список литературы

- [Осовский, 2002] С. Осовский. Нейронные сети для обработки информации – пер. с польского. In: М.: Финансы и статистика, 2002. – 344с.
- [Хайкин, 2001] С. Хайкин. Нейронные сети: полный курс. In: Нейронные сети: полный курс. In: М.: Вильямс, 2006.
- [Alessandri et. al. 2001] Alessandri A., Cuneo M., Pagnan S., Sanguineti M. On the convergence of EKF-based parameters optimization for Neural Networks. In: Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, Hawaii, USA, December 2001, Vol. 6, pp. 6181 – 6186.
- [Bone and Cardot, 2011] Romuald Bone and Hubert Cardot. Advanced Methods for Time Series Prediction Using Recurrent Neural Networks. In: Recurrent Neural Networks for Temporal Data Processing, Chapter 2, InTech, 2011, Croatia.

- [Cernansky and Benuskova, 2003] Cernansky M., Benuskova L. Simple recurrent network trained by RTRL and Extended Kalman Filter algorithms. In: Neural Network World, 2003, No. 13, Vol. 3, pp. 223 – 234.
- [Chernodub, 2012a] A.N. Chernodub. Local Control Gradients Criterion for Selection of Neuroemulators for Model Reference Adaptive Neurocontrol. In: Optical Memory and Neural Networks (Information Optics), 2012, Vol. 21, No. 2, pp. 126-131.
- [Chernodub, 2012b] A.N. Chernodub. Training Neuroemulators Using Multicriteria Extended Kalman Filter and Pseudoregularization for Model Reference Adaptive Neurocontrol. To be appeared in: IV IEEE International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT), St. Petersburg, Russia, October 3 – 5, 2012.
- [De Jesus and Hagan, 2007] De Jesus O., Hagan M.T. Backpropagation: Algorithms for a Broad Class of Dynamic Networks In: IEEE Transactions on Neural Networks. – 2007. – N 1, Vol. 18. – pp. 14 – 27.
- [Gers et. al., 2001] F.A. Gers, D. Eck, J. Schmidhuber. Applying LSTM to Time Series Predictable Through Time-Window Approaches. In: Proceedings of International Conference on Artificial Neural Networks. Berlin, 2001, pp. 669 – 676.
- [Hochreiter and Schmidhuber, 1997] S. Hochreiter, J Schmidhuber. Long Short-Term Memory. In: Neural Computation, 1997, Vol. 9, No. 8, pp. 1735-1780.
- [Hochreiter et. al, 2001] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In: IEEE Press, 2001.
- [Prokhorov et. al., 2001] D. Prokhorov, G. Puskorius, and L. Feldkamp. Dynamical Neural Networks for Control. In: J. Kolen and S. Kremer (Eds.) A Field Guide to Dynamical Recurrent Networks, IEEE Press, 2001.
- [Prokhorov, 2008] Prokhorov D.V. Toyota Prius HEV Neurocontrol and Diagnostics. In: Neural Networks, 2008, No. 21, pp. 458 – 465.

Информация об авторах



Чернодуб Артем – *IEEE Member, младший научный сотрудник отдела Нейротехнологий Института Проблем Математических Машин и Систем НАНУ.*
Адрес: Украина, г. Киев, ул. академика Глушкова 42, ИПММС НАНУ; e-mail:
achernodub@immsp.kiev.ua
Научные интересы: нейронные сети, теория управления, интеллектуальная обработка изображений, биометрическая идентификация.