

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 13:00:36
Уникальный программный идентификатор:
65ab2aa0d384efe8480e6a4c688eddbc475e411a

МИНОБРАЗОВАНИЯ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования «Юго-Западный государственный университет» (ЮЗГУ)

Кафедра программной инженерии



Утверждаю:

Проректор по учебной работе

Локтионова О.Г.

2021г.

**ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ С
ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ**

Методические указания для выполнения лабораторной работы по дисциплине «Теория нейрокомпьютерных систем» для студентов направления подготовки 09.03.04 «Программная инженерия»

Курск 2021

УДК 004.932

Составитель: Р.А. Томакова

Рецензент

Кандидат технических наук, к.т.н., доцент А.В. Малышев

Прогнозирование временных рядов с использованием нейронных сетей: методические указания для проведения лабораторных работ и выполнения самостоятельной внеаудиторной работы по дисциплине «Теория нейрокомпьютерных систем» для студентов всех форм обучения направления подготовки бакалавров 09.03.04 «Программная инженерия»/ Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2021. – 10 с.

Рассмотрена методика прогнозирования временных рядов с использованием нейронных сетей, приведено исследование различных моделей ИНС в системе **MATLAB** с использованием графического инструмента пользователя **NNTOOL** из пакета **Neural Network Toolbox**.

Методические указания соответствуют требованиям рабочей программы дисциплины «Теория нейрокомпьютерных систем».

Предназначены для студентов всех форм обучения направления подготовки бакалавров 09.03.04 «Программная инженерия».

Текст печатается в авторской редакции

Подписано в печать

Формат 60×84 1/16.

Усл. печ. л. 0,9 Уч.- изд. л. 0,8 Тираж 25 экз. Заказ. 883

Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

Цель работы:

- изучение метода прогнозирования временных ряда с помощью нейронных сетей в системе MATLAB;
- приобретение навыков и умений построения прогнозных решений для формирования временных ряда с помощью нейронных сетей в системе MATLAB с использованием графического инструмента пользователя **NNTOOL** из пакета **Neural Network Toolbox**.

ОСНОВНЫЕ ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ. ПОНЯТИЕ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ

Ряд наблюдений $x(t_1), x(t_2), \dots, x(t_N)$ анализируемой величины $x(t)$, произведенных в различные моменты времени t_1, t_2, \dots, t_N , называется *временным рядом* (ВР). В общем виде моменты наблюдений могут быть произвольными, но принято считать, что во временных рядах *производятся равноотстоящие моменты наблюдений*.

Постановка задачи одношагового прогнозирования: имеется временной ряд Y , представленный своими значениями, лагами (ЛАГ от англ. lag - запаздывание, временной лаг – показатель, отражающий отставание или опережение во времени одного явления по сравнению с другими) в предшествующие моменты времени Y_{t-k}, Y_{t-k+1}, Y_t , где t -текущий момент времени, k -глубина выборки.

Необходимо найти величину изменения ее значения, которое произойдет в следующий момент времени, на основе анализа прошлых значений, а также истории изменения других факторов, влияющих на динамику прогнозируемой величины.

При этом важным может являться как максимально низкая величина ошибки прогнозирования, так и правильное определение характера дальнейшего изменения значения прогнозируемой величины (увеличение или уменьшение).

При решении задачи прогнозирования требуется найти значение Y (или несколько величин, вектор) в будущий момент

времени на основе известных предыдущих значений $\vec{Y}, \vec{V}, \vec{C}$:

$$Y_{t+1} = g(\vec{Y}, \vec{V}, \vec{C})$$

$$Y_{t+1} = g(Y_{t-k}, \dots, Y_t, V_{1-t-k}, \dots, V_{n_t}, C_{1-t-k}, \dots, C_{1_t}, \dots, C_{m_{t-k}}, \dots, C_{m_t}) \quad (1)$$

Выражение (1) является основным соотношением, определяющим прогноз.

Определение структуры нейронной сети

Определение структуры нейронной сети предполагает задание ее архитектуры: *число слоев, число нейронов на каждом слое, а также вид функции активации нейронов.*

При использовании многослойного перцептрона для решения задач аппроксимации функций (в т.ч. и задач прогнозирования) нет необходимости в использовании числа скрытых слоев больше одного.

Таким образом, предлагается построить нейронную сеть, имеющую в своем составе входной, *один скрытый и выходной слою*. Число нейронов первого (то есть входного) слоя $n^{(1)}$ зависит от количества входных факторов и глубины ретроспективной выборки по каждому фактору и определяется по формуле:

$$n^{(1)} = \sum_p P \cdot rk_p, \quad (2)$$

где P - число входных факторов, k_p - глубина ретроспективной выборки по p фактору.

Таким образом, *число нейронов входного слоя равно числу элементов входных данных.*

Выходной слой содержит один нейрон, выход которого представляет собой полученное в результате работы нейронной сети значение прогноза.

При необходимости получения одновременно нескольких выходных значений (например, котировок продажи и покупки ценной бумаги) предпочтительнее не увеличивать число выходных нейронов, а обучать несколько отдельных нейронных сетей, каждая из которых имеет один выход.

В пользу предлагаемого подхода свидетельствуют результаты

исследований. Так, в частности, в результате экспериментов было установлено, что *точность прогноза нейронной сети с двумя выходами хуже, чем использование прогнозных решений на основе двух нейронных сетей с единственным выходом при обучении на одних и тех же данных.*

Число нейронов второго (скрытого) уровня $n^{(2)}$ первоначально предлагается принимать равным полусумме числа нейронов входного и выходного слоев:

$$n^{(2)} = \frac{\sum P \cdot k_p + 1}{2}. \quad (3)$$

В процессе определения и настройки параметров нейронной сети число скрытых нейронов может варьироваться с целью увеличения точности прогноза. При этом порядок их количества можно оценить с помощью теоремы Колмогорова и следствия из нее.

В соответствии с этими теоретическими результатами непрерывную функцию можно аппроксимировать с любой точностью при помощи трехслойной нейронной сети, которая имеет N входных, $2 \cdot N + 1$ скрытых и один выходной нейрон.

Таким образом, число $2 \cdot N + 1$ теоретически можно считать верхней границей для числа нейронов на скрытых уровнях. Однако данная оценка справедлива для наилучшей сети, соответствующей глобальному минимуму адаптивного рельефа ошибки в пространстве весов.

На практике достичь глобального минимума обычно не удастся. Кроме того, имея $2 \cdot N + 1$ скрытых нейронов, сеть способна в точности запомнить обучающие примеры. В то же время в реальности от нее требуется их обобщить, выявив скрытые закономерности.

Таким образом, оценка Колмогорова по сути сделана для другой задачи и в данном случае служит лишь ориентиром. В ходе экспериментов эмпирическим путем было *установлено, что число скрытых нейронов может составлять от $N/2$ до $3N$.*

В качестве функции активации нейронов предлагается использовать *биполярную сигмоидальную функцию*. Сигмоидальная функция часто используется в нейронных сетях типа «многослойный перцептрон», так как она является гладкой,

дифференцируемой, а ее производная выражается через саму функцию, что важно для обучения сети.

Однако *рациональная сигмоидальная функция с областью определения $D(f) \in [0, 1]$ имеет недостаток*, связанный с ее несимметричностью относительно оси абсцисс, который способствует уменьшению скорости обучения нейронной сети.

Поэтому в данной лабораторной работе предлагается применять биполярную сигмоидальную функцию. Эксперименты показали, что область определения биполярной функции не оказывает существенного влияния на процесс обучения нейронной сети. Для определенности в дальнейшем будет использоваться функция с областью определения $D(f) \in [-1, 1]$.

Формирование обучающих выборок

Для обучения нейронной сети необходимо сформировать из массива информативных признаков обучающие выборки, каждая из которых содержит вход и требуемый выход нейросети, который сеть должна научиться выдавать в ответ на данный вход.

При этом входные данные содержат историю предыдущих значений по факторам, которые применяются при построении прогноза, а в качестве требуемого выхода используется будущее значение прогнозируемой величины (на этапе обучения оно известно).

В качестве иллюстрации ниже приведен фрагмент реальных данных, и на их основе сформирован обучающий пример.

Таблица 1 - Исходные данные

Название фактора	Значение нормированной котировки на дату					Требуемый выход сети
	04.06.97	05.06.97	06.06.97	09.06.97	10.06.97	
US Treas-5	0	-1.552	0.789	-0.052	0.29	
DowJones			-1.304	-0.393	-.058	
ОГВВЗ-5	-0.124	-0.124	-0.358	-0.245	0	0.462

Таблица 2- Вид сформированного примера

0	-1.552	0.789	-0.052	0.29	-1.304	-0.393	-0.058	-0.124	-0.124	-0.358	-0.245	0	0.462
---	--------	-------	--------	------	--------	--------	--------	--------	--------	--------	--------	---	-------

Как видно, в обучающем примере исходные данные выстраиваются в строку вида *<вход нейросети><выход нейросети>*

В данном случае в этой таблице первые пять значений соответствуют– US Treasuries 5, затем последующие три значения – DowJones, потом пять значений ОГВВЗ 5 и завершает строку значение требуемого выхода сети (то есть будущая котировка ОГВВЗ 5).

Количество нейронов входного слоя сети соответствует числу элементов входных данных, содержащихся в обучающем примере, а число нейронов выходного слоя – соответственно числу эталонных выходов (в данном случае – один).

Совокупность обучающих примеров создается из массива данных на основе информации о значениях, полученных в предшествующие моменты времени с помощью метода «скользящего окна».

Для формирования обучающего примера на массив исходных данных накладывается временное «окно». При этом данные о значениях входных факторов, попавшие в него, формируют обучающий пример.

Следующий пример формируется путем сдвига «окна» на один временной интервал вперед. В качестве ширины «окна» по каждому фактору используется ранее определенная глубина ретроспективной выборки.

Сформированные примеры разделяются на два подмножества: *обучающее и контрольное*.

Обучающая выборка используется для непосредственного обучения сети.

Контрольная выборка – для обучения не используется и служит для оценки способности нейронной сети работать с неизвестными для нее данными.

Обычно число примеров обучающего множества относится к числу примеров контрольного множества как 2:1 или 3:1.

Пример выполнения прогнозирования временного ряда с использованием среды MATLAB

1. В окне “CommandWindow” среды MATLAB введите команду `nntool`. Откроется окно графического инструмента пользователя `nntool`.
2. Нажмите кнопку “New”. Откроется окно создания нейронной

сети.

3. В текстовом поле ввода “Name” введите наименование нейронной сети, например, nntpredict.

4. В выпадающем списке “NetworkType” выберите позицию “Feed-forward backprop”.

5. В текстовом поле ввода “Numbersoflayers” введите количество слоев нейронной сети равное двум (скрытый слой и выходной слой).

6. Так, как в качестве входных данных для обучения будем использовать данные, представленные в таблице 2, количество нейронов скрытом слое (текстовое поле ввода “Numbersofneurons”) зададим равным 39 (мощность множества входных данных, умноженное на 3).

7. Откройте вкладку “Data” и задайте входные данные $P = [0; -1.552; 0.789; -0.052; 0.29; -1.304; -0.393; -0.058; -0.124; -0.124; -0.358; -0.245; 0]$.

8. Задайте результат прогноза $T = [0.462]$.

9. Откройте вкладку “Network”. В выпадающем списке “Inputdata” выберите позицию “P”. В выпадающем списке “Targetdata” выберите позицию “P”.

10. Нажмите кнопку “View”. Откроется окно, содержащее модель структуры нейронной сети (рис. 1).

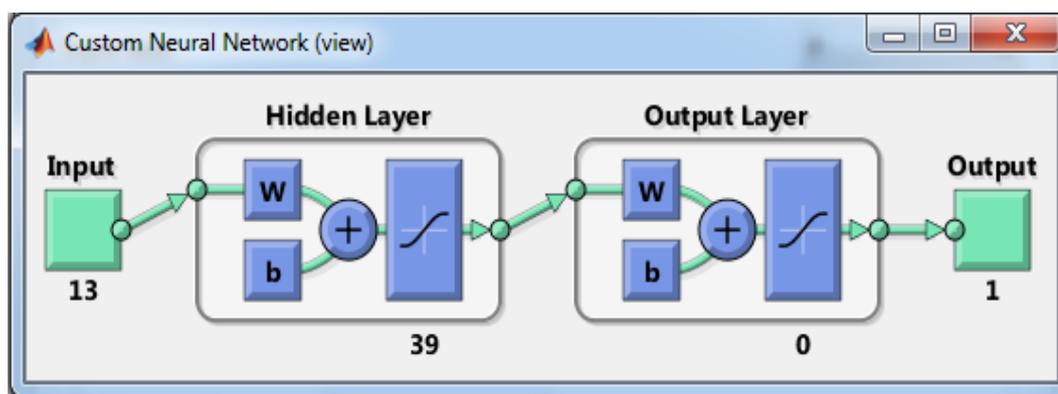


Рисунок 1- Модель структуры нейронной сети

11. Закройте окно, содержащее модель структуры нейронной сети. В окне создания нейронной сети нажмите кнопку “Create”. Закройте окно создания нейронной сети.

12. В окне графического инструмента пользователя nntool дважды кликните мышкой по наименованию нейронной сети.

13. Откроется окно обучения и тестирования нейронной сети.

14. Откройте вкладку “Train”. В выпадающем списке “Inputs” выберите позицию “P”. В выпадающем списке “Targets” выберите позицию “T”. Нажмите кнопку “TrainNetwork”. Откроется окно обучения нейронной сети. Дождитесь окончания обучения и закройте окно обучения нейронной сети.
15. Откройте вкладку «Simulate». В выпадающем списке “Inputs” выберите позицию “P”. Нажмите кнопку “SimulateNetwork”. Дождитесь окончания симуляции нейронной сети.
16. Закройте окно обучения и тестирования нейронной сети.
17. Просмотрите результат симуляции сети.
18. Сделайте выводы.

Таблица 3- Вид временного ряда и диапазон его изменения

Вариант	Вид функции	Промежуток изменения
1	$(1.85-t)*\cos(3.5t-0.5)$	$t \in [0,10]$
2	$\cos(\exp(t))$	$t \in [2,4]$
3	$\sin(t)*t$	$t \in [3.1,20]$
4	$\sin(2t)*t$	$t \in [3.1,20]$
5	$\cos(2t)*t$	$t \in [2.3,20]$
6	$(t-1)\cos(3t-15)$	$t \in [0,10]$
7	$\cos(3t-15)$	$t \in [1,10]$
8	$\cos(3t-15)/\text{abs}(t)=0$	$t \in [0,0.3),(0.3,10]$
9	$\cos(3t-15)*t$	$t \in [0,1]$
10	$(\exp(t)-\exp(-t))\cos(t) / (\exp(t)+\exp(-t))$	$t \in [0,5]$
11	$(\exp(-t)-\exp(t))\cos(t) / (\exp(t)+\exp(-t))$	$t \in [0,5]$
14	$\cos(t-0.5)*\text{abs}(t)$	$t \in (0,10]$
15	$\cos(2t)*\text{abs}(t-2)$	$t \in (2,10]$

Контрольные вопросы

1. Что такое временной ряд?
2. С помощью какой нейронной сети выполняется прогнозирование временного ряда?
3. Как формируется архитектура нейронной сети?
4. Как определяется число слоев скрытых нейронов?
5. Сформулируйте принципы формирования обучающей выборки.

6. Какие особенности содержат принципы формирования проверочной выборки?

7. Какие особенности содержат принципы формирования обучающих выборок?

8. В чем заключается идея метода «скользящего окна»?

9. Сформулируйте постановку задачи одношагового прогнозирования.

10. Что характеризует функция активации нейронов, как она выбирается?

11. Перечислите виды функций активации, в чем заключаются их особенности?

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Основные теоретические сведения об исследуемом процессе.
4. Временной ряд, построенный с помощью функции, соответствующей варианту задания.
5. Обучающая выборка.
6. Тестовые выборки.
7. Результаты обучения и тестирования нейронной сети.
8. Выводы.