***Ключевые слова**:*
искусственные нейронные сети,
кластеризация данных, роевые методы обучения,
сеть прямого прохода,
карта самоорганизации (сеть Кохонена)

***С. С. Головачев**, аспирант*
кафедры международных валютно-финансовых
отношений НИУ-ВШЭ
*(**e-mail: serg_golovachev@mail.ru**)*

Кластеризация данных и роевые методы обучения искусственных нейронных сетей в прогнозировании рынка ценных бумаг

Мировой финансовый рынок — это сложная, быстро меняющаяся динамическая система. Поэтому для эффективного прогнозирования его колебаний зачастую недостаточно использовать стандартные модели и приходится разрабатывать более сложные алгоритмы, включающие в себя сразу несколько теоретических подходов. В рамках данной статьи рассмотрено использование искусственных нейронных сетей — широко применяемого сегодня аналитического инструмента для прогнозирования фондового рынка на долгосрочном временном промежутке.

В качестве базовой модели возьмем простую сеть прямого прохода (*feed-forward artificial neural network*) и осуществим достройку модели с целью повышения точности результатов ее работы.

Для повышения эффективности базовой нейронной сети применим роевой механизм обучения (*swarm intelligence*) и введем несколько независимых нейронных сетей, которые будут конкурировать друг с другом. Принципиальное нововведение относительно других роевых моделей — использование особого набора входных данных. Эти данные будут получены путем сегментирования исходного временного ряда на основе расстояния Дженсона-Шеннона и их последующего сведения в кластеры. Еще одним относительно новым механизмом будет адаптивная подстройка весов на основе карты самоорганизации (сети Кохонена).

Для наглядной демонстрации эффективности работы нашей модели сравним динамику виртуального инвестиционного портфеля, построенного по ее прогнозам, с рыночной динамикой и динамикой виртуального портфеля, построенного на основе прогноза авторегрессионной модели.

ВЫБОР МОДЕЛИ

Сеть прямого прохода широко описана во многих работах, посвященных нейросетевой тематике[1]. Мы выбираем данную нейронную сеть, потому что она достаточно проста в построении, и это поможет нам: — представить каждую сеть прямого прохода как независимый идентичный объект (именно к таким группам объектов применяется роевой механизм обучения); — легко оценить приращение в эффективности по мере модификации и усложнения модели.

Наша сеть будет иметь два нейрона во входном слое, два нейрона в скрытом слое и один — в выходном. В качестве функции активации возьмем стандартную логистическую функцию. Обучение сети будет происходить по методу наименьших квадратов (МНК)[2].

Таким образом, можно записать нашу нейронную сеть как функцию ошибки, которую мы будем стремиться свести к минимуму на скользящем[3] периоде, который в данном случае будет равен пяти торговым дням:

$$f(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,z) = \sum_{i=z-5}^{z-1}\left[\left[\frac{1}{1+e^{-\left(\varepsilon\frac{1}{1+e^{-\left(\alpha r_{1,z}+\beta r_{2,z}\right)}}+\zeta\frac{1}{1+e^{-\left(\gamma r_{1,z}+\delta r_{2,z}\right)}}\right)}}\right]-r_{z-1}\right]^2 \rightarrow Min, \quad (1)$$

где z — номер торгового дня;

$r_{1,z}$, $r_{2,z}$ — эмпирические входные данные для сети с определенным лагом[4];

α, β, γ, δ, ε, ζ — веса сети, которые оптимизируются в процессе обучения по МНК.

Указав основные параметры базовой модели, мы можем перейти к ее доработке на основе: 1) кластеризации входных данных; 2) применения роевых механизмов обучения; 3) адаптивной подстройки весов с помощью карты самоорганизации (сеть Кохонена), что будет описано в следующих частях статьи.

ПРИНЦИПЫ РОЕВОГО ОБУЧЕНИЯ

Для совершенствования исходной модели попытаемся внедрить в нее элементы роевого обучения. Алгоритмы роевого поведения заимствованы из природы, где

---

[1] *См., напр. Kröse B., van der Smagt P. An Introduction To Neural Networks (1996).*

[2] *Подробнее об архитектуре сети прямого прохода см.: Головачев С. С. Использование искусственных нейронных сетей для прогнозирования американского фондового рынка в период кризиса (2012).*

[3] *Скользящий период (скользящее окно) означает, что при расчете прогноза для каждого последующего торгового дня мы также на один день сдвигали «вправо» по временной оси наши входные значения. Благодаря этому искусственные нейронные сети постоянно обучались на актуальной информации. Это крайне важно для прогнозирования финансовых рынков, где данные стремительно устаревают и становятся нерелевантными.*

[4] *В качестве эмпирического массива данных выбраны ежедневные логарифмические значения доходностей американского фондового индекса S&P 500 за период с 1 марта 2005 г. по 19 ноября 2012 г. (около 1800 значений — достаточно длинный временной ряд, с помощью которого можно получить объективную оценку эффективности работы нашей модели).*

группы животных (стаи, рои) демонстрируют неожиданно хорошие результаты при том, что каждая из особей обладает довольно примитивным разумом.

В последние несколько десятилетий в различных областях науки делаются небезуспешные попытки воссоздать роевые механизмы поведения в искусственных моделях, и финансовая математика не исключение[5]. Многие модели при этом стремятся воспроизвести свойство, присущее природным самоорганизующимся системам. Система в процессе взаимодействия ее элементов как целое приобретает качества, которые не были присущи элементам по отдельности[6].

При построении нашей модели мы также хотели бы использовать эти полезные свойства. Вначале зададим набор «особей» — отдельных, независимых нейронных сетей, — которые затем будут конкурировать между собой и повышать общую эффективность модели (конкуренция и естественный отбор — основополагающие эволюционные процессы в природе).

Для этого потребуются одинаковые нейронные сети, описанные в первой части данной статьи. При этом будет некорректно изменять число нейронов в них или тип функции активации, иначе мы в неявном виде определим преимущество той сети, чьи базовые «настройки» окажутся наиболее адекватными ситуации на рынке. Однако в этом случае нейронные сети будут выдавать идентичные результаты, и никакой конкуренции между ними не будет. Мы предлагаем оригинальный выход из данной ситуации: обучать каждую нейронную сеть на отдельном временном ряду, вычлененном из исходного массива данных.

Для этой цели мы на каждом торговом шаге z будем работать со скользящим массивом из 200 значений логарифмической доходности индекса S&P 500. Внутри этого массива будет проводиться сегментирование данных, а затем объединение их в кластеры. В дальнейшем каждая из нейронных сетей будет обучаться на своем (не пересекающемся с другими) кластере.

СЕГМЕНТИРОВАНИЕ И КЛАСТЕРИЗАЦИЯ ДАННЫХ

Сегментирование данных будет производиться в соответствии с расстоянием Дженсона-Шеннона, которое является средним расстоянием (энтропией) Кульбака-Ляйблера и часто используется для оценки различия между несколькими распределениями[7].

Формально расстояние Дженсона-Шеннона (*JS*) можно задать следующим образом:

$$JS(p_1, p_2) = \frac{KL(p_1(x), p_2(x)) + KL(p_2(x), p_1(x))}{2}, \quad (2)$$

где p_1, p_2 — нормальные распределения;
KL — расстояние Кульбака-Ляйблера между этими двумя распределениями, которое определяется как

$$KL(p_1(x), p_2(x)) = \sum_i \ln\left(\frac{p_1(i)}{p_2(i)}\right) p_1(i). \quad (3)$$

Сама процедура сегментирования может быть представлена следующим образом[8]:

---

[5] *См., напр. Kendall G., Su Y. A Particle Swarm Optimisation Approach In The Construction of Optimal Risky Portfolios (2005); Nenortaitė J. A Particle Swarm Optimization Approach In The Construction of Decision-Making Model (2007); Majhi R., Panda G., Sahoo G. Panda A., Choubey A. Prediction of the S&P 500 and DJIA Stock Indices using Particle Swarm Optimization Technique.*

[6] *Cont R., Bochaude J.-P. Herd Behavior and Aggregate Fluctuations In Financial Markets (2000); Baraviera A.T., Bazzan A.L.C., da Silva R. Emerging Collective Behavior in a Simple Artificial Financial Market (2005).*

[7] *См. Barbea J., Rao C.R. On the convexity of some divergence measures based on entropy functions (1982); Kullback S., Leibler R.A. On Information and sufficiency (1951).*

[8] *Wong J.C., Lian H., Cheong S.A. Detecting macroeconomic phases in the Dow Jones Industrial Average time series (2009).*

1. Мы предполагаем, что исходный массив может быть разбит на *n* — число отрезков (сегментов). Для этой цели задаются два параметра: а) минимальное количество точек, которое будет задавать распределение (в нашем случае оно равно 5), б) пороговое значение расстояния Дженсона-Шеннона, с помощью которого можно будет проводить различия между сегментами (распределениями) — в нашем случае оно будет равно 2.

2. После этого проводим итеративную процедуру деления исходного 200-точечного массива данных на сегменты. На первой итерации массив разбивается надвое. Точку разделения определяем на основе расстояния Дженсона-Шеннона, измеряемого между исходным распределением и если бы оно было представлено смесью из двух нормальных распределений слева и справа от этой точки:

$$JS(q) = \log \frac{p_2(q)}{p_1}, \tag{4}$$

где *JS*(*q*) — расстояние Дженсона-Шеннона;
q — точка разделения исходного массива;
p_1 — функция правдоподобия исходного нормального распределения

$$p_1 = \prod_{t=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{5}$$

где *N* — длина исходного массива (в нашем случае 200 точек);
μ, σ — параметры нормального распределения p_1;
$p_2(q)$ — функция правдоподобия для смеси из двух нормальных распределений слева и справа от точки разделения *q*:

$$p_2(q) = \prod_{t=1}^{q} \frac{1}{\sigma_L\sqrt{2\pi}} e^{-\frac{\left(x-\mu_L\right)^2}{2\sigma_L^2}} \cdot \prod_{t=q+1}^{N} \frac{1}{\sigma_R\sqrt{2\pi}} e^{-\frac{\left(x-\mu_R\right)^2}{2\sigma_R^2}}, \tag{6}$$

где μ_L, σ_L — параметры нормального распределения слева от точки разделения *q*;
μ_R, σ_R — параметры нормального распределения справа от точки разделения *q*.

3. На последующих итерациях каждая из двух получившихся частей разбивается надвое, затем — каждая из четвертей, восьмых и так далее до тех пор, пока не будет выполнено одно из условий из п.1: либо длина сегмента становится равна минимально установленному размеру распределения, либо расстояние Дженсона-Шеннона становится ниже порогового значения, т. е. распределение и смесь из двух его частей уже невозможно отличить друг от друга в информационном плане.

После того, как процедура сегментирования завершена, необходимо «склеить» получившиеся распределения в кластеры. Сведение нескольких групп данных в кластеры и определение их оптимального числа — нетривиальная задача, в решении которой могут быть использованы различные подходы.

В рамках нашей модели применим метод *k*-средних[9]. Каждый из получившихся сегментов может быть нанесен на плоскость в виде точки с двумя координатами — параметрами распределения (среднее значение и стандартное отклонение). После этого начинаем группировать точки вокруг центров кластеров, определяя центры так, чтобы суммарное квадратичное отклонение точек от центров было минимальным.

Выбор количества кластеров производится эвристическим способом. Мы последовательно увеличиваем число кластеров до тех пор, пока качество кластеризации не перестанет существенно улучшаться[10]. Под качеством кластеризации мы понимаем

[9] *См. Steinhaus H. Sur la division des corps materiels en parties (1956).*
[10] *См., напр. Thorndike R. L. Who belongs in the family? / Psychometrika, 1953.*

сумму расстояний от точек кластеров до центров (S) и, соответственно, будем следить за ее убыванием по мере добавления новых кластеров:

$$S = \sum_{i=1...c} \sum_{j=1...n} f(u_i, K_{i,j}), \quad (7)$$

где *c* — число кластеров;
n — число точек в каждом кластере;
u_i — центр *i*-того кластера;
$K_{i,j}$ — координаты точки в кластере;
$f(u_i, K_{i,j})$ — функция расстояния до центров кластеров.

По итогам процедуры кластеризации на каждом скользящем 200-точечном массиве было выявлено, что число кластеров, при котором качество кластеризации перестает существенно возрастать, равно пяти[11].

На основе этого результата на следующем шаге зададим точно такое же число искусственных нейронных сетей (особей), которые будут конкурировать между собой. При этом следует отметить, что средние длины кластеров отличаются друг от друга не более чем на 20 %, из чего можно сделать допущение, что каждая из нейронных сетей будет работать с информационной выборкой примерно одинаковой мощности.

Кроме этого, проведенный анализ кластеров показал, что они распределены внутри общего массива достаточно равномерно, т. е. явные концентрации данных одного типа на отдельных сегментах (например, в начале или в конце массива) отсутствуют.

ВНЕДРЕНИЕ РОЕВОГО МЕХАНИЗМА ОБУЧЕНИЯ НА КЛАСТЕРИЗОВАННЫХ МАССИВАХ ДАННЫХ

После того, как каждый скользящий массив из 200 логарифмированных значений доходности фондового индекса S&P 500 был разбит на 5 различных кластеров с примерно одинаковой информационной мощностью, мы создали ровно такое же число искусственных нейронных сетей-близнецов. Это точно такие же сети прямого прохода, что и описанные в (1), с той разницей, что в рамках каждого скользящего 200-точечного массива они будут обучаться не на 5-дневном отрезке, а на всех данных, включенных в отдельный кластер (длины кластеров могут варьироваться в зависимости от торгового дня *z*).

Еще раз заметим, что мы берем за основу достаточно простую нейронную сеть, чтобы оценить эффект от применения кластеризации данных и внедрения роевого механизма обучения, а не от сложных взаимосвязей внутри конструкций самой сети.

Теперь можно перейти непосредственно к применению роевого механизма обучения. По итогам обучения на кластерах в каждой сети будут определены оптимальные веса. На основе этих весов каждая из пяти сетей на каждом торговом шаге *z* будет рассчитывать свой тестовый виртуальный инвестиционный портфель в течение 50 торговых дней, следующих за исходным 200-дневным скользящим массивом.

Виртуальный инвестиционный портфель формировался в соответствии со следующим торговым правилом. В случае, если абсолютное значение прогноза сети в торговый день *z* было больше или равно некоему пороговому значению *T*, мы вставали в длинную позицию, в противном случае — в короткую (8). В качестве порогового значения T может выступать как некая константа, взятая из области значения функции активации нейронной сети, или как определенная переменная. Например, мы можем попытаться принимать решение на основе сравнения прогноза сети с ее собственным значением на один или несколько временных шагов назад или с лагированным значением доходности индекса S&P 500.

Следует отметить, что мы намеренно старались использовать достаточно примитивное торговое правило, чтобы, по возможности, минимизировать влияние самого правила на общий результат работы модели.

---

[11] *Автор выражает благодарность А. В. Исакову за помощь в проведении расчетов.*

Формально наше торговое правило выглядит так:

$$G_z = \begin{cases} r_z & \text{если } \zeta_z \geq T \\ -r_z & \text{иначе} \end{cases}, \quad (8)$$

где G_z — логарифмическая доходность виртуального инвестиционного портфеля в торговый день z (чтобы вернуться к обычным доходностям, мы применяли экспоненциальную функцию для вычисления динамики виртуального инвестиционного портфеля;
r_z — логарифмическая доходность индекса S&P 500 в торговый день z;
ξ_z — скалярное значение прогноза нейронной сети для торгового дня z;
T — пороговое значение для прогноза.

Мы сознательно применяли одинаковые торговые правила для всех пяти сетей, чтобы обеспечить максимально честные «правила игры» и не предопределить выигрыша какой-либо из них. Также акцентируем внимание на том, что тестовый виртуальный инвестиционный портфель определялся не единожды, но тоже был скользящим и соответствовал каждому из 200-точеченых скользящих начальных массивов, из которых вычленялись кластеры.

По итогам 50-дневной тестовой торговли среди нейронных сетей в рое определялся победитель на основе величины виртуального инвестиционного портфеля. Таким образом происходила конкуренция между сетями и возникал определенный эволюционный отбор наиболее эффективной модели.

Однако в рамках данной модели мы решили пойти несколько дальше и «передавать» качества сети-«победителя» остальным сетям. Данный подход находится на стыке роевых механизмов обучения и генетических алгоритмов, когда определенные гены скрещиваются, мутируют и передаются от «родителей» к «потомкам».

После подстройки весов других сетей под веса сети-«победителя» повторим тестовую торговлю на том же 50-дневном периоде. Мотив для осуществления этой операции заключается в том, что, по нашему предположению, в ходе обучения на кластерах сети прямого прохода могли найти локальный минимум ошибки прогноза, и повторная «прогонка» с улучшенными весами на том же массиве позволит прийти к глобальному минимуму. На практике это будет означать, что после повторной тестовой торговли «победителем» может оказаться другая сеть.

ИСПОЛЬЗОВАНИЕ КАРТЫ САМООРГАНИЗАЦИИ (СЕТИ КОХОНЕНА)

Подстройку весов остальных сетей под значения весов сети-«победителя» мы будем осуществлять с помощью добавления в модель еще одной нейронной сети — карты самоорганизации (сети Кохонена). Следует подчеркнуть, что это важное новаторство данной работы, поскольку в рамках одной модели мы соединяем сеть прямого прохода, которая использует традиционный механизм обучения по МНК («обучение с учителем»), роевые и эволюционные методы обучения, а также самообучающуюся нейронную сеть, которой является карта самоорганизации.

Карта самоорганизации (сеть Кохонена) в плане архитектуры представляет собой несколько иную структуру, чем сеть прямого прохода. По сути, в ней речь идет о различных отображениях исходных векторов-сигналов в пространстве выходного слоя, который имеет большую размерность. Построение отображений в выходном слое делается с целью составления т. н. карты признаков (систем координат) исходного вектора, где каждое отображение должно содержать в себе значимую информацию о какой-то характеристике исследуемого объекта[12].

В качестве весов, которые обычно оптимизируются при работе с искусственными нейронными сетями, в картах самоорганизации будут выступать группы координат, которые задают отображения исходного вектора в выходном пространстве.

---

[12] *Хайкин С. Нейронные сети: полный курс (2006); Kohonen T. Self-organized formation of topologically correct feature maps (1997) и др.*

Отметим, что сеть Кохонена — это самообучающаяся искусственная нейронная сеть, которую также можно рассматривать как сигнальную систему: отображения исходного вектора (или векторов) конкурируют между собой, в результате чего и происходит активизация одного из нейронов. Процесс ее оптимизации представляет собой встраивание других отображений в топологическую окрестность вокруг победившего нейрона, отсюда и название сети — карта самоорганизации. Процесс определения упорядоченной топологической окрестности вокруг отображения-«победителя» называется процессом кооперации, а подстройка других отображений под свойства данной топологической области — процессом адаптации.

При использовании сети Кохонена в нашей модели будем считать, что поиск победившего отображения уже выполнен. Мы определили отображение-«победитель» (в нашем случае веса отдельной нейронной сети из роя) на основе величины тестового виртуального инвестиционного портфеля по итогам 50-дневной торговли, и теперь можем перейти непосредственно к процессам кооперации и адаптации сети Кохонена.

Особо подчеркнем, что обычно под «отображениями» подразумеваются веса внутри одной сети Кохонена, однако в данном случае мы предположили, что конкуренция происходит между несколькими независимыми нейронными сетями (роем). Это, безусловно, следует отнести к одному из уникальных качеств рассматриваемой в статье модели.

Пошагово процессы кооперации и адаптации сети Кохонена могут быть представлены следующим образом:

1. Формируется топологическая окрестность вокруг сети-«победителя», которая определяет то, каким образом веса других сетей должны измениться в зависимости от весов сети-«победителя» (процесс кооперации). Топологическую окрестность мы будем находить на основе Евклидова расстояния между весами сети-«победителя» и весами остальных сетей. Здесь следует отметить, что существуют и другие подходы к определению этого расстояния, например, на основе величины относительной энтропии по Кульбаку-Ляйблеру, которая упоминалась ранее. Полученные Евклидовы расстояния мы будем использовать в качестве аргументов функции Гаусса (9), которая будет экспоненциально убывать во времени (т. е. на каждой последующей итерации). Отметим, что именно использование Гауссовой функции обеспечивает нелинейное преобразование в сети Кохонена:

$$h_n^j = e^{\frac{-\left[\left| w_{n-1}^j - w_{n-1}^{win} \right|\right]^2}{2\left(\sigma e^{\frac{-n}{\tau_0}}\right)^2}}, \qquad (9)$$

где h_n^j — функция топологической окрестности;
w — веса одной из нейронных сетей из (1),
j — номер сети;
n — номер итерации, на которой осуществляется подстройка весов;
win — номер сети-«победителя», по весам которой корректируются веса других сетей;
σ — стандартное отклонение Гауссова распределения;
τ_0 — временная константа.

2. Производится подстройка весов в других сетях под веса сети-«победителя» с помощью следующей итеративной процедуры обучения (процесс адаптации):

$$w_n^j = w_{n-1}^j + \eta \cdot e^{\frac{-n}{\tau_1}} \cdot h_n^j \cdot \left[X - w_{n-1}^j \right], \qquad (10)$$

где X — эмпирический вектор из исходного 200-точечного массива данных с определенным лагом ко дню прогноза;
w — веса одной из нейронных сетей из (1);
j — номер сети;
n — номер итерации, на которой осуществляется подстройка весов;
h_n^j — функция топологической окрестности;

η — параметр, влияющий на скорость обучения. Одно из условий стохастической аппроксимации (разновидностью которой является сеть Кохонена) — это зависимость параметра обучения от времени, что обеспечивается с помощью множителя на основе экспоненциальной функции (в нашем случае параметр обучения будет убывающим);
τ_1 — временная константа.

После того, как процесс адаптации весов других сетей из роя завершен, мы вновь осуществляем тестовую торговлю на том же скользящем 50-дневном периоде и на основе величины нового тестового портфеля опять определяем победившую нейронную сеть. Полученные результаты говорят о том, что практически в 100% случаев после подстройки весов в сети Кохонена номер сети-«победителя» менялся. Это служит убедительным доказательством того, что в ходе первой тестовой торговли мы, скорее всего, находили лишь локальный минимум ошибки прогноза, а после внедрения процессов кооперации и адаптации сети Кохонена приходили в глобальный минимум. В свою очередь, это должно существенно повысить прогнозную эффективность нашей итоговой модели, которую мы проанализируем в следующей части статьи.

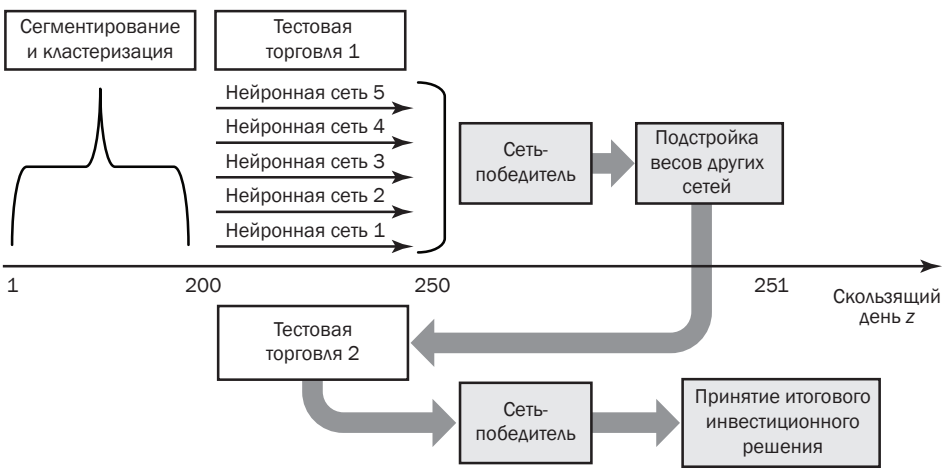
РЕЗУЛЬТАТЫ РАБОТЫ МОДЕЛИ

Прежде чем перейти к обзору результатов работы модели еще раз кратко пошагово зафиксируем наш алгоритм (рис. 1):

1. Было произведено сегментирование и кластеризация исходных эмпирических данных (еще раз заметим, что обучающая выборка постоянно смещается, и мы выполняем процедуру сегментирования и кластеризации для каждого торгового дня);
2. Пять идентичных нейронных сетей из (1) провели первый тестовый торговый раунд на скользящем 50-дневном промежутке;
3. Для каждого скользящего 50-дневного промежутка была определена сеть-«победитель» по наибольшей абсолютной величине торгового портфеля;
4. С помощью механизма, заимствованного из карт самоорганизации (сеть Кохонена), на том же 50-дневном скользящем промежутке была осуществлена подстройка весов остальных четырех сетей под веса сети-«победителя»;
5. Те же пять нейронных сетей провели еще один тестовый торговый раунд на том же 50-дневном скользящем промежутке;
6. Вновь для каждого скользящего 50-дневного промежутка была определена сеть-«победитель» по наибольшей абсолютной величине торгового портфеля, причем почти всегда это была не та же сеть, что на шаге 3.

Рисунок 1

Схема работы предложенной модели



Источник: составлено автором.

Заметим, что оба тестовых торговых раунда — на шаге 3 и 6 — проводились по заранее известным данным (прогноз in-the-sample), т. е. они пытались построить аппроксимацию по заранее известным данным. Для реальной торговли это неприменимо, потому что в день *z* – 1 мы никогда не можем знать значение доходности в день *z*. Поэтому итоговый прогноз нашей модели будет формироваться на основе выходного значения сети, победившей на шаге 7, на 1 торговый день вперед за границами обучающей выборки (прогноз out-of-the-sample).

Для обеспечения лучшей сравнимости с другими моделями принятия решений сравним результаты нашей модели с результатами прогноза по простой авторегрессионной модели:

$$r_z^* = \alpha_z + \beta_z r_{z-1}, \quad (11)$$

где r_z^* — прогноз авторегрессионной модели для торгового дня *z*,

α_z, β_z — коэффициенты авторегрессии для торгового дня *z*,

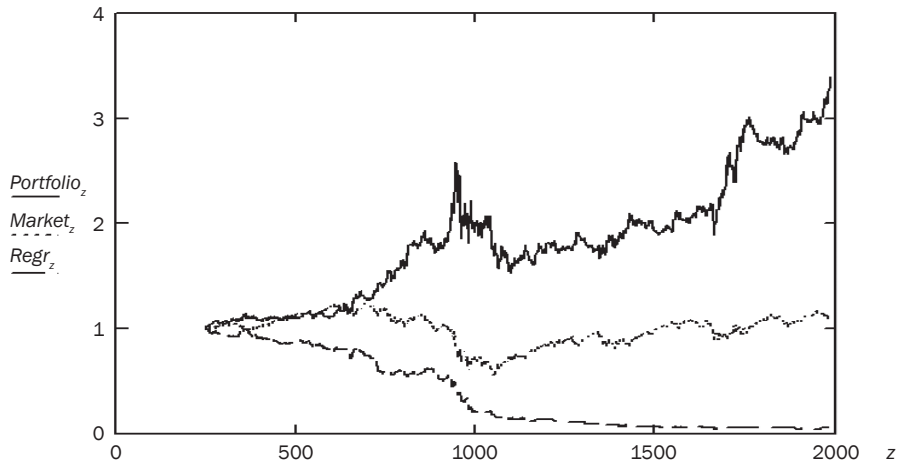
r_{z-1} — логарифмическое значение доходности индекса S&P 500 в день (*z*–1).

Начальное значение портфеля, формируемого на основе прогнозов искусственных нейронных сетей (виртуальный инвестиционный портфель *Portf*), было равно 1. Далее, в зависимости от прогноза сетей, мы ежедневно вставали в длинную либо в короткую позицию, руководствуясь торговым правилом из (8). Начальное значение портфеля на основе авторегрессии (*Regr*) было равно 1. Далее, в зависимости от значения прогноза, мы ежедневно вставали в длинную, либо в короткую позицию, руководствуясь торговым правилом из (8). Под рыночной динамикой мы будем понимать динамику доходности самого индекса S&P 500 во времени (рыночный портфель), при том, что в начале рассматриваемого периода мы принимали его значение равным 1 (*Market*).

Наше предположение состоит в том, что если с помощью торговой стратегии, основанной на использовании искусственных нейронных сетей, нам удастся превзойти динамику рынка и авторегрессионный прогноз, то это будет сигналом того, что данный алгоритм способен с некоторой степенью достоверности воспроизводить поведение действующих на нем агентов.

Рисунок 2

Торговые результаты, полученные с помощью нашей модели, авторегрессии, а также рыночная динамика



Источник: рассчитано автором.

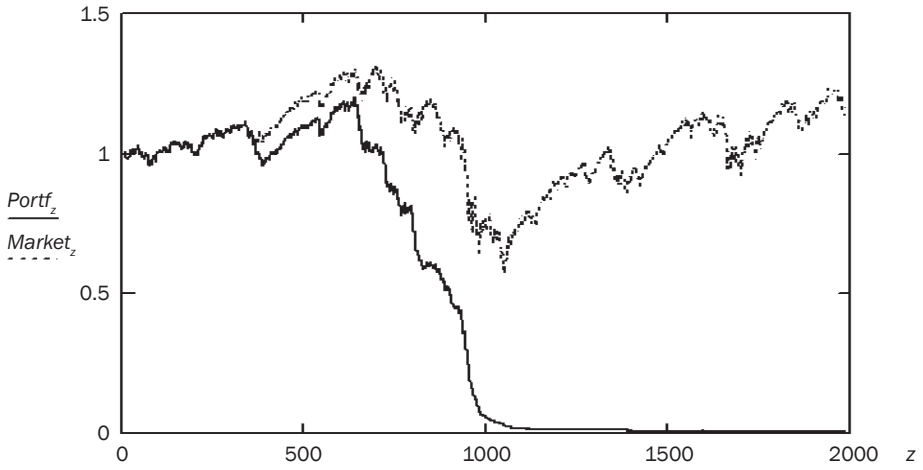
Примечание: по горизонтальной оси отложены торговые дни (z), по вертикальной — величина портфеля. Market — рыночный портфель, Portfolio — виртуальный инвестиционный портфель, построенный на основе прогноза нашей модели, Regr — виртуальный инвестиционный портфель, построенный на основе прогноза авторегрессии. Начальные значения портфелей равны 1.

На конец рассматриваемого периода значение нашего виртуального инвестиционного портфеля составило 3,39, портфеля, построенного по прогнозам авторегрессии — 0,05, а рыночного портфеля — 1,11. Результат нашей модели существенно превосходит и рыночную динамику, и динамику портфеля, построенного в соответствии с авторегрессионным прогнозом.

Очень важно отметить, что благодаря кластеризации исходных данных, использованию роевых механизмов обучения и карт самоорганизации (сеть Кохонена) мы смогли существенно улучшить работу нейронной сети прямого прохода. Если бы мы использовали для прогноза рынка базовую модель, описанную в первой части статьи, то на конец рассматриваемого периода значение рыночного портфеля составило 1,17, а значение виртуального портфеля — практически ноль (см. рис. 3). Это означает, что наш виртуальный инвестиционный портфель полностью проиграл бы «рынку», а реальный инвестор потерял все вложенные средства

Рисунок 3

Торговые результаты, полученные с помощью базовой модели (простая нейронная сеть прямого прохода), авторегрессии, а также рыночная динамика



Источник: рассчитано автором.
Примечание: по горизонтальной оси отложены торговые дни (z), по вертикальной — величина портфеля. Market — рыночный портфель, Portf — виртуальный инвестиционный портфель, построенной на основе прогноза сети прямого прохода. Начальные значения портфелей равны 1.

В качестве дополнительной проверки эффективности нашей модели мы выясним, является ли превышение доходности виртуального инвестиционного портфеля (*Portfolio*) над рыночным (*Market*) на рис. 2 случайным, либо такая динамика закономерна. Если связь между рыночной и портфельной доходностью полностью отсутствует, это будет означать, что наши инвестиционные результаты могли получиться в результате случайного стечения обстоятельств. Напротив, если между двумя рядами существует статистически значимая взаимосвязь, это будет означать, что результаты работы модели в некоторой степени обусловлены реальными эмпирическими данными и их обработкой.

Для выполнения этой задачи мы начнем с расчета коэффициента корреляции Пирсона между временными рядами *Portfolio* и *Market*. Его значение равно 0,66, и он статистически значим на 95% доверительном интервале (*t*-статистика равна 82,02 при *t*-критическом — 1,97, что говорит о необходимости отвергнуть нулевую гипотезу о незначимости коэффициента).

Далее проведем тест причинности Грейнджера[13], с помощью которого выясним, существует ли причинная зависимость ряда Portfolio от ряда Market. Для выполнения теста Грейнджера необходимо построить две регрессии. Первая — простая авторегрессия для временного ряда *Portfolio* с лагом в один день. При этом мы должны подобрать лаги регрессора так, чтобы коэффициент при нем был значимым. В нашем случае *t*-статистика для коэффициента β равна 755,18 при *t*-критическом –1,97, что говорит о необходимости отвергнуть нулевую гипотезу о незначимости коэффициента.

После этого мы строим вторую регрессию, в которую включаем значения временного ряда *Market*. Мы вновь должны подобрать лаг переменной *Market* так, чтобы коэффициент при ней был статистически значимым. В нашем случае *t*-статистика для коэффициента γ равна –2,28 при *t*-критическом –1,97, что говорит о необходимости отвергнуть нулевую гипотезу о незначимости коэффициента.

После того, как обе регрессии построены, мы рассчитываем на основе сумм их квадратичных ошибок F-статистику, которая и должна дать итоговый ответ по тесту Грейнджера. В нашем случае F-статистика равна 58,71, при F-критическом — 3,02. Это говорит о том, что нулевая гипотеза теста Грейнджера о том, что временной ряд *Market* причинно не обуславливает временной ряд Portfolio, должна быть отвергнута.

Таким образом, можно с уверенностью утверждать, что существует значимая статистическая взаимосвязь между реальной рыночной доходностью и нашим виртуальным инвестиционным портфелем. Это означает, что положительная динамика портфеля относительно рынка не может быть признана случайной.

Итак, мы получили еще одно подтверждение и можем с уверенностью говорить, что наша модель может быть успешно использована для прогнозирования фондового рынка на долгосрочном временном интервале.

Кроме этого необходимо выделить и другой, более важный с научной точки зрения, результат. Успех нашей модели в прогнозировании реальной рыночной динамики говорит о том, что, в соответствии с нашими исходными предположениями, мы смогли сконструировать механизм, который с достаточно высокой точностью воспроизводит алгоритм принятия решений индивидуумами, действующими на рынке. Это еще раз подтверждает перспективность использования искусственных нейронных сетей для построения моделей, призванных имитировать реальные биохимические процессы в голове человека.

ЗАКЛЮЧЕНИЕ

В рамках данной работы мы пытались прогнозировать динамику американского фондового рынка с помощью модели, основанной на искусственных нейронных сетях. Взяв за основу искусственную нейронную сеть прямого прохода, мы модифицировали модель.

Полученные эмпирические результаты (динамика виртуального инвестиционного портфеля) демонстрируют, что нейросетевая модель, основанная на использовании роевых механизмов обучения и адаптивной подстройки весов по методу карты самоорганизации (сети Кохонена), может эффективно прогнозировать динамику рынка в долгосрочной перспективе, обыгрывая как рыночный портфель, так и портфель, построенный на основе авторегрессионного прогноза.

Более важный результат, чем потенциальный инвестиционный успех торговой стратегии, основанной на использовании построенной модели, заключается в том, что искусственные нейронные сети такого типа могут с достаточной степенью точности

[13] *См. Granger C. W. J. Investigating Causal Relations by Econometric Models and Cross-spectral Methods (1969).*

воспроизводить механизмы поведения и принятия решений участниками фондового рынка. Только в случае успеха подобной имитации наша модель была бы успешна в инвестиционном плане. Дальнейшую доработку модели имеет смысл сосредоточить, например, на использовании других типов и числа нейронных сетей, а также построить прогнозы для других временных промежутков (часовые, минутные котировки) и других типов финансовых рынков (например, валютного или фьючерсного).

Библиография

1. Головачев, С. С. Использование искусственных нейронных сетей для прогнозирования американского фондового рынка в период кризиса // Электронный научный журнал «Управление экономическими системами». — Режим доступа: http://www.uecs.ru/instrumentalnii-metody-ekonomiki/item/1644-2012-11-09-08-08-12).
2. Исаков, А. В. Сигнальная модель для внутреннего денежного рынка // Прикладная эконометрика. — 2013. — № 29 (2).
3. Круглов, В. В., Борисов, В. В. Искусственные нейронные сети: Теория и практика. — М.: Горячая линия-Телеком, 2002 — 382 с.: ил.
4. Хайкин, С. Нейронные сети: полный курс, 2-е издание / Пер. с англ. — М.: Издательский дом Вильямс, 2006 — 1104 с.
5. Bishop, C. M. Neural Networks for Pattern Recognition // Oxford University Press, 1995. — 483 p.
6. Baraviera, A. T., Bazzan A. L. C., da Silva R. Emerging Collective Behavior in a Simple Artificial Financial Market // S´lvio R. Dahmennstituto de F´sica, UFRGS, 2005.
7. Barbea, J., Rao, C. R. On the convexity of some divergence measures based on entropy functions / IEEE Trans. Information Theory. — Vol. 28. — 1982.
8. Bonabeau, E., Sobkowski, A., Theraulaz, G., Deneubourg, J.-L. Adaptive Task Allocation Inspired by a Model of Division of Labor in Social Insects, 1998.
9. Cont, R., Bochaude, J.-P. Herd Behavior and Aggregate Fluctuations In Financial Markets // Macroeconomic Dynamics. — 2000. — P. 170–196.
10. Granger, C. W. J. Investigating Causal Relations by Econometric Models and Cross-spectral Methods // Econometrica. — 1969. — 37 (3). — P. 424–438.
11. Kendall, G., Su, Y. A Particle Swarm Optimisation Approach In The Construction of Optimal Risky Portfolios / Proceedings of the 23rd IASTED International Multi-Conference Artificial Intelligence and Applications. — Innsbruck, Austria. — 2005. — February 14-16.
12. Kohonen, T. Self-organized formation of topologically correct feature maps // Bio1ogical Cybernetics. — 1982. — vol. 43. — P. 59-69.
13. Kohonen, T. The self-organizing map / Proceedings of the Institute of Electrical and Electronics Engineers. — 1990. — vol. 78. — P. 1464-1480.
14. Kohonen, T. Exploration of very large databases by self-organizing maps / Interactional Conference on Neural Networks. — 1997. — vol. 1. — P. PL1-PL6, Houston.
15. Kullback, S., Leibler, R. A. On Information and sufficiency // Annals of Mathematical Statistics. — 1951. — 22 (1). — P. 79–86.
16. Kröse, B., van der Smagt, P. An Introduction To Neural Networks / Eight Edition. — The University of Amsterdam, November 1996.
17. Majhi, R., Panda, G., Sahoo, G. Panda, A., Choubey, A. Prediction of the S&P 500 and DJIA Stock Indices using Particle Swarm Optimization Technique / IEEE Congress on Evolutionary Computation, 2008.
18. Nenortaitė, J. A Particle Swarm Optimization Approach In The Construction of Decision-Making Model // Information Technology and Control. — 2007. — Vol. 36. — № 1A.
19. Steinhaus, H. Sur la division des corps materiels en parties / Bull. Acad. Polon. Sci., 1956. — C1. III vol IV: 801—804.
20. Wong, J. C., Lian, H., Cheong, S. A. Detecting macroeconomic phases in the Dow Jones Industrial Average time series / Physica A, 2009.
21. Thorndike, R. L. Who belongs in the family? / Psychometrika, 1953.