

---

---

## АНАЛИЗ ЭФФЕКТИВНОСТИ НЕЧЕТКИХ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧЕ ОЦЕНКИ КРЕДИТНОГО РИСКА

Шовгун Наталия

**Abstract:** *The problem of assessing the creditworthiness of the borrower is considered. The application of fuzzy neural networks for this problem solution was suggested. The experimental investigations of the adaptation and learning algorithms of fuzzy neural networks were carried out. The modification of adaptation and learning algorithms of fuzzy neural networks was suggested.*

**Keywords:** *fuzzy neural networks, credit rating, fuzzy logic, genetic algorithm*

**ACM Classification Keywords:** *H.4 Information systems applications - H.4.2. Types of Systems Decision Support*

---

### Введение

Управление кредитным риском осуществляется на индивидуальном и портфельном уровне. На портфельном уровне происходит процесс оценки совокупного риска портфеля ссуд, а на индивидуальном - оценка риска связанного с несостоятельностью заемщика. В оценке таких характеристик заемщика, как финансовое состояние и моральные качества, содержится доля субъективизма. Кроме того оценка кредитоспособности может различаться от банка к банку. Информация, на основе которой принимается решение о кредитоспособности, может быть неточной и неполной, а данные о заемщике могут быть трудно формализуемы. Для учета этих факторов целесообразно использовать подход на основе нечеткой логики. Однако, недостатком данного подхода является то, что системы, построенные на основе нечеткого логического вывода, не способны автоматически обучаться, а параметры и вид функций принадлежности, которые описывают факторы кредитоспособности, выбираются экспертом и могут оказаться не адекватными. Устранить этот недостаток можно с помощью нечетких нейронных сетей (ННС), которые способны к адаптации и моделированию сложных процессов. При этом существует возможность интерпретации полученных результатов на основе базы правил полученной в процессе настройки ННС. В работе рассматриваются ННС с выводом Мамдани и Сугено [Зайченко, 2008], а также каскадная нео-фаззи сеть [Бодянский, 2011]. Для ННС с выводом Сугено проводится анализ эффективности методов обучения сети и предлагается использовать комбинацию генетического и градиентного методов обучения. Также предлагается алгоритм адаптации ННС, в котором настраивается число правил из базы знаний, а не только параметры функций принадлежности.

---

### Нечеткая нейронная сеть с выводом Сугено

В нечеткой нейронной сети выводы делаются на основе аппарата нечеткой логики, но соответствующие функции принадлежности подстраиваются с использованием алгоритмов обучения нейронных сетей, например, алгоритма обратного распространения ошибки. Такие системы не только используют априорную информацию, но могут приобретать новые знания и для пользователя являются логически прозрачными. Рассмотрим нечеткую нейронную сеть с выводом Сугено.

В сети TSK (Takagi, Sugeno, Kang'a) используется такая база правил [Зайченко, 2008]:

$R_1$ : если  $x_1$  это  $A_1^{(1)}$ ;  $x_2$  это  $A_2^{(1)}$ ; ...;  $x_n$  это  $A_N^{(1)}$ , то  $y_1 = p_{10} + \sum_{j=1}^N p_{1j}x_j$ ,

$R_M$ : если  $x_1$  это  $A_1^{(M)}$ ;  $x_2$  это  $A_2^{(M)}$ ; ...;  $x_n$  это  $A_N^{(M)}$ , то  $y_M = p_{M0} + \sum_{j=1}^N p_{Mj}x_j$ ,

где  $A_i^{(k)}$  - значение лингвистической переменной  $x_i$ ,  $i=1,2...N$  (данные из анкеты заемщика) для правила  $R_k$  с функцией принадлежности (ФП)  $\mu_A^{(k)}(x_i) = \frac{1}{1 + \left(\frac{x_i - c_i^{(k)}}{\sigma_i^{(k)}}\right)^{2b_i^{(k)}}}$ ,  $M$  – количество правил.

В ННС TSK выделяют 5 слоев:

1. Первый слой выполняет раздельную фаззификацию каждой переменной  $x_i$ , определяя для каждого  $k$ -го правила вывода значение ФП  $\mu_A^{(k)}(x_i)$  в соответствии с функцией фаззификации, например, колоколообразной. Это – параметрический слой с параметрами  $c_j^{(k)}$ ,  $\sigma_j^{(k)}$ ,  $b_j^{(k)}$ , которые подлежат настройке в процессе обучения сети.

2. Второй слой выполняет агрегирование отдельных переменных  $x_i$ , определяя результирующую степень принадлежности  $w_k$  вектора  $X$  условиям  $k$ -го правила, где

$$w_k = \mu_A^{(k)}(x) = \prod_{j=1}^N \frac{1}{1 + \left(\frac{x_j - c_j^{(k)}}{\sigma_j^{(k)}}\right)^{2b_j^{(k)}}}.$$

3. В третьем слое рассчитываются значения  $y_k(x) = p_{k0} + \sum_{j=1}^N p_{kj}x_j$  (реализуется нечеткий алгоритм

Сугено), а также производится умножение функций  $y_k(x)$  и  $w_k$ , сформированных в предыдущем слое. Это параметрический слой с параметрами  $p_{k0}, p_{kj}$ , которые подлежат настройке.

4. В четвертом слое рассчитывается взвешенная сумма сигналов  $y_k(x)$  и сумма весов  $\sum_{k=1}^M w_k$ .

5. В пятом слое веса подлежат нормализации и вычисляется выходной сигнал  $y(x) = \frac{\sum_{k=1}^M w_k y_k(x)}{\sum_{k=1}^M w_k}$ .

В градиентном методе происходит минимизация целевой функции ошибки  $E = \frac{1}{L} \sum_{l=1}^L (y_l - d_l)^2$ , где  $L$  – объем обучающей выборки. Формулы обучения градиентным методом для ФП имеют вид:

$$c_j^{(k)}(n+1) = c_j^{(k)}(n) - n_c \frac{\partial E(n)}{\partial c_j^{(k)}},$$

$$\sigma_j^{(k)}(n+1) = \sigma_j^{(k)}(n) - n_\sigma \frac{\partial E(n)}{\partial \sigma_j^{(k)}},$$

$$b_j^{(k)}(n+1) = b_j^{(k)}(n) - n_b \frac{\partial E(n)}{\partial b_j^{(k)}}.$$

При настройке параметров ФП  $c_j^{(k)}$ ,  $\sigma_j^{(k)}$ ,  $b_j^{(k)}$  можно использовать алгоритм Resilient Propagation [Riedmille, 1992] для сокращения процесса обучения сети.

Параметры  $p_{k0}$ ,  $p_{kj}$  находятся в результате решения системы из L линейных уравнений, при фиксации

прочих параметров. Система имеет вид  $d^l = \frac{\sum_{k=1}^M w_k y_k(x^l)}{\sum_{k=1}^M w_k}$  при фиксации прочих параметров, где  $d^l$  -

желаемый выход сети.

В генетическом методе обучения все настраиваемые переменные составляют особь популяции:

$$W = [c_1^{(1)}, \dots, c_N^{(1)}, b_1^{(1)}, \dots, b_N^{(1)}, \sigma_1^{(1)}, \dots, \sigma_N^{(1)}, p_{10}, p_{11}, \dots, p_{1N}, c_1^{(M)}, \dots, c_N^{(M)}, b_1^{(M)}, \dots, b_N^{(M)}, \sigma_1^{(M)}, \dots, \sigma_N^{(M)}, p_{M0}, p_{M1}, \dots, p_{MN}]$$

1. Задается начальная популяция из L особей.

2. Вычисляется индекс пригодности  $FI(W_i) = C - \frac{1}{L} \sum_{l=1}^L (y_l - d_l)^2$ , где C – константа.

3. Выбираются особи для скрещивания вероятностным механизмом. Вероятность выбора i-го родителя

$$P_i = \frac{FI(W_i)}{\sum_{l=1}^L FI(W_l)}.$$

4. Далее происходит скрещивание.

5. Популяция подвергается действию мутации.

6. Формируется новая популяция.

Градиентный алгоритм сильно зависит от начальных условий, поэтому предлагается применять гибридный алгоритм обучения, в котором начальное приближение (начальные значения параметров ФП), близкое к оптимальному, находится с помощью генетического алгоритма, а уже потом оно рассматривается как отправная точка для алгоритма градиентного спуска.

Для решение задачи также применялась каскадная нео-фаззи сеть. В этой сети в качестве узлов используется нео-фаззи нейрон, который реализует отображение  $\hat{y} = \sum_{i=1}^n f_i(x_i) = \sum_{i=1}^n \sum_{j=1}^h w_{ji} \mu_{ji}(x_i)$ , где

$\mu_{ji}$  - треугольная функции принадлежности (ФП) входной переменной,  $w_{ji}$  - синаптический вес.

Параметры ФП задаются заранее, например, с помощью дискретизации входных данных. Алгоритм обучения детально описан в [Бодянский, 2011].

Для анализа кредитоспособности заемщика с помощью предложенных методов использовалась выборка данных одного из украинских банков, содержащая 998 кредитных заявок за 2010г. Результаты экспериментов приведены в таблице 1. Количество правил в базе знаний выбрано равным 5.

Табл.1 Результаты оценки кредитоспособности разными методами

Метод	СКО	Процент ошибочных классификаций, %
ННС TSK с генетическим алгоритмом обучения	0.129	13.7
ННС TSK с градиентным алгоритмом обучения	0.116	13.0
ННС TSK с гибридным алгоритмом обучения	0.115	12.3
Нео-фаззи каскадная сеть	0.102	14.7
ННС с выводом Мамдани с генетическим алгоритмом обучения	0.187	15.0
ННС с выводом Мамдани с градиентным алгоритмом обучения	0.112	14.7

Также для сравнения разных методов использовались ROC-кривые методов (зависимость доли верно классифицированных положительных ответов от общего числа положительных классификаций от доли неверно классифицированных отрицательных ответов от общего числа положительных классификаций). Чем ближе кривая к верхнему левому углу, тем выше предсказательная способность модели.

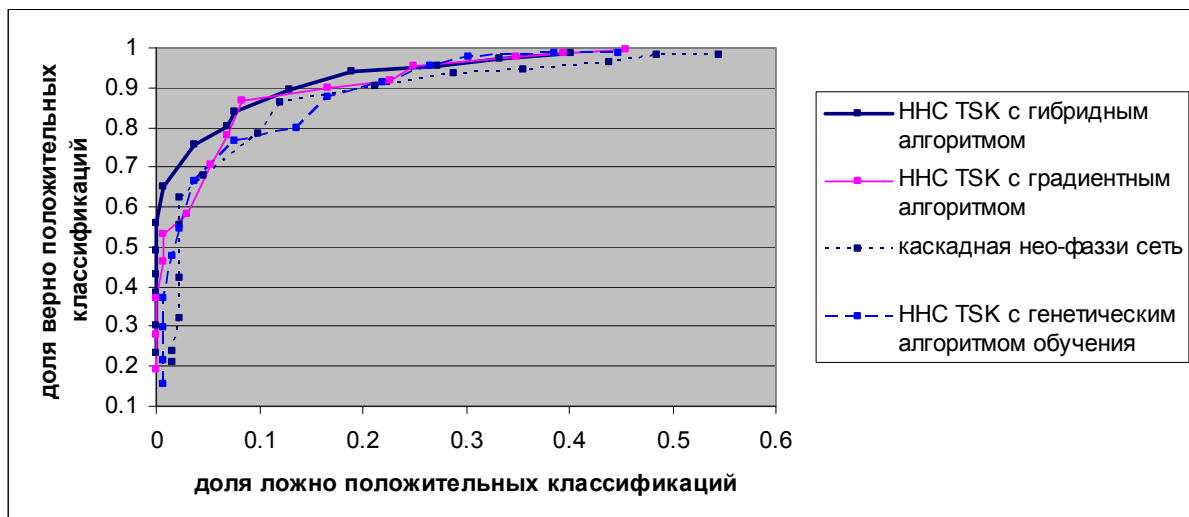


График 1. ROC-кривые методов

Как видно из графика 1 наиболее качественно работает ННС TSK с гибридным алгоритмом обучения.

### Алгоритм адаптации нечетких нейронных сетей

База правил составленных экспертами может быть противоречивой или неполной. В таких ситуациях предпочтительнее использовать алгоритм адаптации нечетких нейронных сетей, который настраивает не только параметры функций принадлежности, но и структуру сети – составляется база правил и соответственно определяется количество нейронов второго и третьего слоя. В алгоритме описаном в [Jaung, 1999] новое правило добавляется если выполняется условие  $I = \arg \max_{1 \leq k \leq r} F^k(x) \leq F_{in}$ , где  $F^k(x(t)) = w_k$  - активирующая сила правила k для образца t, а  $F_{in}$  - выбранный заранее порог, который уменьшается в процессе обучения. Для повышение эффективности предлагается делать еще

одну проверку - новое правило добавляется если выполняется условие  $f_o = \min(f_1, f_2, \dots, f_m) > R$ , где  $f_k = \|\bar{c}^{(k)} - \bar{x}^i\|$ ,  $k = 1, m$  - расстояние между центрами ФП каждого правила и рассматриваемой точкой,  $R = const$ . В сгенерированном правиле параметры ФП выбираются следующим образом:

$$c_j^{(k)} = x_j(t), \quad b_j^{(k)} = 1, \quad \sigma_j^{(k)} = \beta \cdot \prod_{j=1}^N \frac{1}{1 + \left(\frac{x_j - c_j^{(l)}}{\sigma_j^{(l)}}\right)^{2b_j^{(l)}}}. \quad \text{Для сравнения использовалась}$$

модификация алгоритма адаптации, предложенного в [Круглов, 2002]. В случае когда известны все обучающие точки, алгоритм имеет вид:

1. Задаются начальные значения:  $\varepsilon$  - погрешность аппроксимации,  $m = m_0$  - число правил в базе знаний;
2. Выбирается очередная точка  $\bar{x}^i$ , для которой значение  $|d^i - y^i|$  максимальное. Если база знаний пуста, то переходим на шаг 5. Иначе с помощью алгоритма нечеткого логического вывода Сугено и существующей базы правил рассчитывается значение  $d^i$ .
3. Проверяется выполнение неравенства  $f_o = \min(f_1, f_2, \dots, f_m) > R$ , где  $f_k = \|\bar{c}^{(k)} - \bar{x}^i\|$ ,  $k = 1, m$  - расстояние между центрами ФП каждого правила и рассматриваемой точкой,  $R = const$ . Если оно не выполняется, то переходим на шаг 6.
4. Далее проверяется условие  $|d^i - y^i| \leq \varepsilon$  - оценивается точность аппроксимации для существующей базы правил. При его невыполнении переходим на шаг 5, иначе - на шаг 6.
5. В базу знаний добавляется правило с такими значениями ФП:  $c_j^{(k)} = x_j^i$ ,  $b_j^{(k)} = 1$ ,

$$\sigma_j^{(k)} = \beta \cdot \prod_{j=1}^N \frac{1}{1 + \left(\frac{x_j - c_j^{(o)}}{\sigma_j^{(o)}}\right)^{2b_j^{(o)}}}. \quad \text{Количество правил равно } m = m + 1.$$

6. Проверяется правило останова: просмотрены ли все обучающие точки. Если нет, то выбирается следующая обучающая точка.

Использовалась ННС TSK с гибридным алгоритмом обучения. Результаты для этих алгоритмов приведены в таблице 2.

Если экспертом задано избыточное число правил в базе знаний, то в процессе обучения, некоторые правила будут повторяться. Но структура сети будет соответствовать предложенному числу правил – будет излишне сложной. Поэтому автоматическая настройка правил дает возможность построения ННС с оптимальной структурой.

Надо заметить, что количество сгенерированных правил регулируется параметрами  $F_{in}, R$  для алгоритма, основанного на активирующей силе правил, и параметрами  $\varepsilon, R$  для алгоритма, основанного на оценке точности аппроксимации. Результаты экспериментов для разных значений параметров представлены на графиках 2-5.

Как видно из графиков 2-3 в алгоритме адаптации, основанном на активирующей силе правила при фиксированном значении  $R$ , число правил увеличивается при росте  $F_{in}$  до 0.5. При фиксированном значении  $F_{in}$ , чем больше  $R$ , тем меньше правил генерируется. Как видно из графиков 4-5 в алгоритме

адаптации, основанном на оценки точности аппроксимации при фиксированном значении  $R$ , число правил уменьшается при росте  $\varepsilon$ . Это значит, чем меньшую точность мы требуем от нашей модели, тем меньшее число правил создается. При фиксированном значении  $\varepsilon$ , чем больше  $R$ , тем меньше правил генерируется. Параметр  $R$  влияет на число сгенерированных правил в том смысле, что чем большее расстояние между правилами мы требуем, тем меньше правил создается.

Табл.2 Результаты для алгоритмов адаптации

Метод	Количество правил	СКО	Процент ошибочных классификаций, %
ННС TSK с алгоритмом, основанном на активирующей силе правил	5	0.125	14.7
	3	0.119	10.7
ННС TSK с алгоритмом, основанном на оценки точности аппроксимации	2	0.112	12.3
	5	0.117	13.3

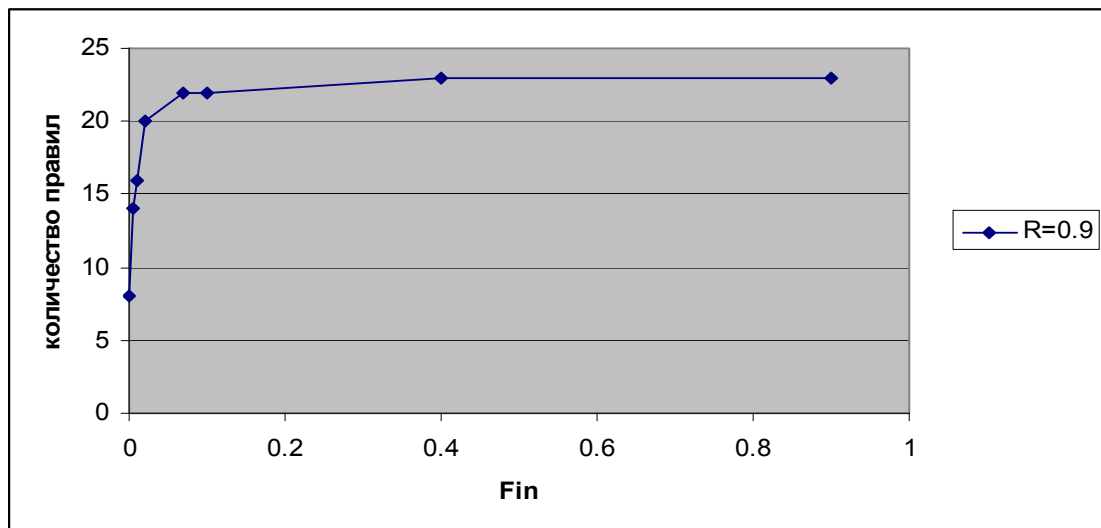


График 2.

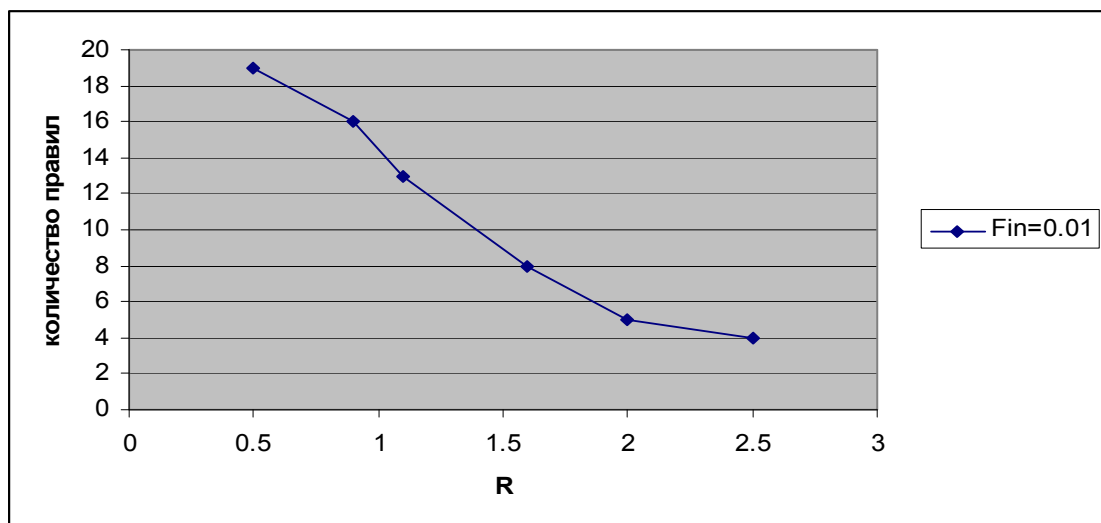


График 3.

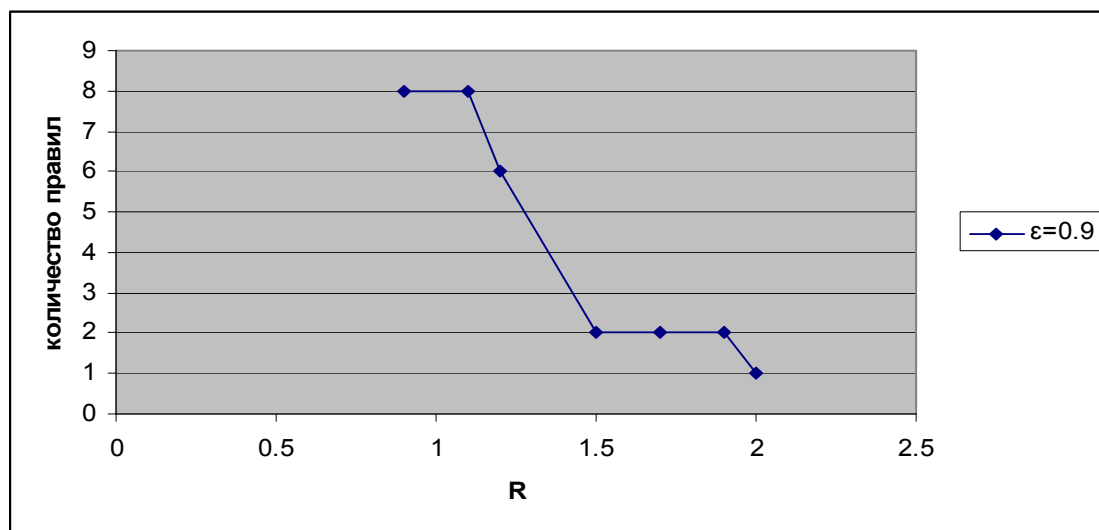


График 4.

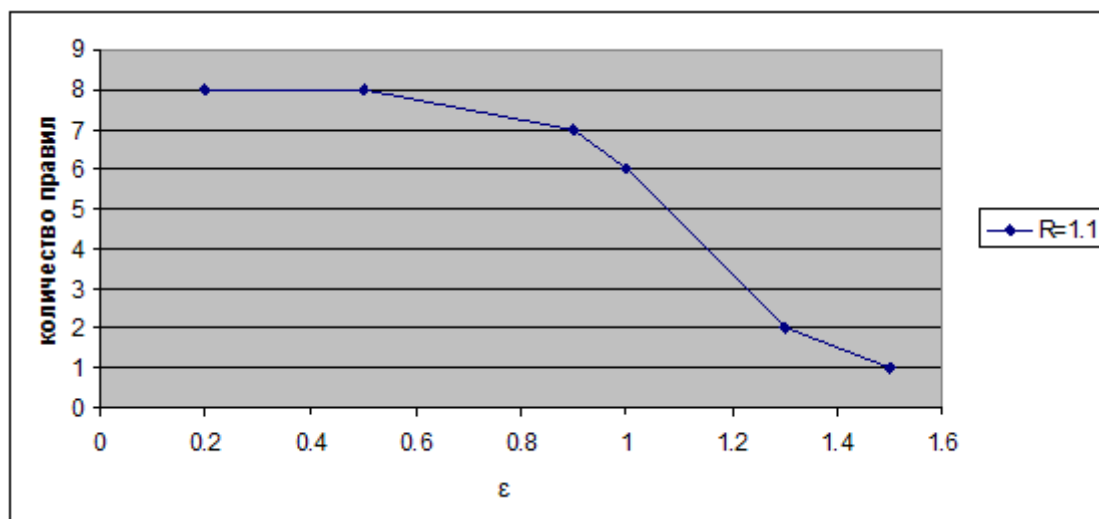


График 5.

### Кредитный риск на портфельном уровне

На портфельном уровне происходит процесс оценки совокупного риска портфеля ссуд. Кредитный портфель банка формируется на основе критерия максимизации дохода от кредитных операций, с учетом ограничения на выделяемые денежные ресурсы.

$\max(F(n_i, R_i))$ , при ограничениях

$$g(n_i) \leq K, n_i \in \{0,1\},$$

где  $K$  – лимит кредитования,  $g(n_i)$  – выданные кредиты,  $F(n_i, R_i)$  – доходность портфеля,  $R_i$  – уровень кредитоспособности заемщика,  $n_i$  – решение о выдаче кредита.

---

Доходность по выданным ссудам зависит от уровня кредитоспособности заемщика  $R_i$ , который может задаваться, например, нечетким треугольным числом. Таким образом, получаем задачу нечеткого математического программирования с нечеткой целевой функцией. Задача оценки кредитного риска на портфельном уровне в нечеткой постановке может быть предметом дальнейших исследований.

---

### Выводы

Проведен анализ эффективности методов обучения нечетких нейронных сетей, применяемых в задаче оценки кредитного риска. Полученные результаты показали эффективность использования, при обучении ННС, гибридного алгоритма (комбинация градиентного и генетического методов). Лучшие результаты показала ННС TSK с гибридным алгоритмом обучения. Предложен новый алгоритм адаптации нечеткой нейросети, который заключается в автоматической настройке правил. Как следствие, происходит построение оптимальной структуры сети, для обучения которой необходимо меньшее количество времени. Реализация предложенных алгоритмов позволяет настраивать базу знаний автоматически без привлечения экспертов. Кроме того, предложена нечеткая модель оценки кредитного риска на портфельном уровне.

---

### Благодарности

*"The paper is published with financial support by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA (www.ithea.org ) and the Association of Developers and Users of Intelligent Systems ADUIS Ukraine ( www.aduis.com.ua )."*

---

### Литература

- [Juang, 1999] CF Juang and CT Lin A recurrent self-organizing neural fuzzy inference network// IEEE transaction of neural networks, Volume 10, Number 4, Pages 828–845, July 1999
- [Riedmille, 1992] Riedmiller M., Brawn H. RPROP - a fast adaptive learning algorithms. Technacal Report // Karlsruhe: University Karlsruhe. 1992.
- [Бодянский, 2011] Бодянский Е.В. Каскадная эволюционная нейронная сеть с нео-фаззи-нейронами в качестве узлов// Восточно-Европейский журнал передовых технологий. – 2011. – Вып. 4/3 (52). – С. 55–58.
- [Зайченко, 2008] Зайченко Ю.П. Нечеткие модели и методы в интеллектуальных системах. – К.: Издательский дом «Слово», 2008. – 334с.
- [Зайченко, 2008] Зайченко Ю.П. Оценка кредитных банковских рисков с использованием нечеткой логики// Intelligent Information and Engineering Systems. – 2008. - №13 – с. 190-200
- [Круглов, 2002] Круглов В.В., Борисов В. В. Искусственные нейронные сети. Теория и практика. -2-е изд., стереотип.- М.: Горячая линия-Телеком, 2002. -382 с

---

### Информация об авторе

**Шовгун Наталия Виталиевна**, аспирантка Национального технического университета Украины «КПИ», адрес электронной почты: [shovgun@gmail.com](mailto:shovgun@gmail.com)