

УКРАЇНА



# СВІДОЦТВО

про реєстрацію авторського права на твір

№ 89692

Комп'ютерна програма "Програмне забезпечення для моделювання статичних нейронних мереж"

(вид, назва твору)

Автор(и) Якушенко Олександр Сергійович, Мільцов Володимир Євгенійович, Корольов Петро Васильович, Охмакевич Валерій Миколайович, Мірзосє Азєр Джаваншір огли, Абдуллаєв Парвіз Шахмурад огли

(повне ім'я, псевдонім (за наявності))

Дата реєстрації

11.06.2019



Державний секретар Міністерства економічного розвитку і торгівлі України **О. Ю. Перевезенцев**



**МІНІСТЕРСТВО ЕКОНОМІЧНОГО РОЗВИТКУ І ТОРГІВЛІ УКРАЇНИ  
(Мінекономрозвитку)**

вул. М. Грушевського 12/2 м. Київ 01008 тел. 253-93-94, факс 253-63-71  
Web: <http://www.me.gov.ua>, e-mail: [meconomy@me.gov.ua](mailto:meconomy@me.gov.ua), код згідно з ЄДРПОУ 37508596

№ \_\_\_\_\_ На № \_\_\_\_\_ від \_\_\_\_\_

**Мільцов В.Є.**

**вул. Ніжинська, 10,  
кв.1,  
м. Київ.,  
03054**

Мінекономрозвитку направляє рішення за заявкою № 90763 від 13.05.2019р. та Свідоцтво про реєстрацію авторського права на твір № 89692 від 11.06.2019р.

Додаток: зазначене на 2 арк. в 1 прим.

**Заступник директора департаменту  
інтелектуальної власності – начальник  
відділу авторського права та суміжних прав **Оксана ЯРМОЛЕНКО****

Івченко О.С.. 596 67 58\*3864#



ДОКУМЕНТ СЕД Мінекономрозвитку АСКОД  
Сертифікат 33B6CB7BF721B9CE0400000050B52600087C5F00  
Підписувач Яроменко Оксана Володимирівна  
Дійсний з 06.02.2018 0:00:00 по 06.02.2020 0:00:00

Мінекономрозвитку



2311-12/30315-09 від 22.07.2019  
17:04:50



**МІНІСТЕРСТВО ЕКОНОМІЧНОГО РОЗВИТКУ І ТОРГІВЛІ УКРАЇНИ  
(Мінекономрозвитку)**

вул. М. Грушевського, 12/2, м. Київ, 01008, тел. (044) 253-93-94, факс (044) 253-63-71  
Web: <http://www.me.gov.ua>, e-mail: [meconomy@me.gov.ua](mailto:meconomy@me.gov.ua), код згідно з ЄДРПОУ 37508596

**Р І Ш Е Н Н Я**

**ПРО РЕЄСТРАЦІЮ АВТОРСЬКОГО ПРАВА НА ТВІР**

Міністерство економічного розвитку і торгівлі України розглянуло заяву

**Мільцов Володимир Євгенійович, вул. Ніжинська, 10, кв. 1, м. Київ, 03054**

(повне ім'я автора, адреса)

заявка від 13.05.2019 № 90763

про реєстрацію авторського права на твір і прийняло рішення зареєструвати авторське право на твір **Комп'ютерна програма "Програмне забезпечення для моделювання статичних нейронних мереж"; Якушенко Олександр Сергійович, Мільцов Володимир Євгенійович, Корольов Петро Васильович, Охмакевич Валерій Миколайович, Мірзосв Азєр Джаваншір огли, Абдуллаєв Парвіз Шахмурад огли**

(вид, повна, скорочена (за наявності) назва твору, повне ім'я, псевдонім (за наявності) автора (ів))

Внесення відомостей до Державного реєстру свідоцтв про реєстрацію авторського права на твір та видача свідоцтва будуть здійснені за умови сплати збору за оформлення і видачу свідоцтва про реєстрацію авторського права на твір відповідно до п.3 постанови Кабінету Міністрів України від 27 грудня 2001 року № 1756 "Про державну реєстрацію авторського права і договорів, які стосуються права автора на твір".

Якщо протягом трьох місяців від дати одержання заявником рішення про реєстрацію авторського права на твір Управління державних реєстрацій Департаменту інтелектуальної власності Міністерства економічного розвитку і торгівлі України не одержало документ про сплату збору за оформлення і видачу свідоцтва у розмірі та порядку, визначених законодавством, або копію документа, що підтверджує право на звільнення від сплати зазначеного збору, заявка вважається відхиленою і реєстрація авторського права та публікація відомостей про реєстрацію Управління державних реєстрацій Департаменту інтелектуальної власності Міністерства економічного розвитку і торгівлі України не проводиться.

Державний секретар  
Міністерства економічного розвитку  
і торгівлі України



О. Ю. Перевезенцев

### **„Програмне забезпечення для моделювання статичних нейронних мереж”**

Програму написано на C++ і призначено для моделювання роботи статичної нейронної мережі з довільною кількістю шарів у мережі та нейронів у кожному шарі. Функції активації: лінійна; логарифмічна; гіперболічний тангенс.



**КЕРІВНИЦТВО КОРИСТУВАЧА**  
**комп'ютерної програми "Програмне забезпечення для моделювання статичних**  
**нейронних мереж"**

Розробники:

Якушенко Олександр Сергійович

Корольов Петро Васильович

Мільцов Володимир Євгенійович

Охмакевич Валерій Миколайович

## ЗМІСТ

1.	ЗАГАЛЬНИЙ ОПИС ПРОГРАМНОГО ПРОДУКТА.....	7
2.	ОПИС ФАЙЛУ ДАНИХ ДЛЯ ІНІЦІАЛІЗАЦІЇ НЕЙРОНОЇХ МЕРЕЖІ.....	8
3.	ОПИС КЛАСІВ, З ЯКИХ СКЛАДАЄТЬСЯ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ .....	8
4.	ПРИКЛАД РОБОТИ З ПРОГРАМНИМ ПРОДУКТОМ .....	13
4.1.	Приклад файлу ініціалізації.....	13
4.2.	Приклад створення, ініціалізації та використання статичної нейронної мережі.....	13

## 1. ЗАГАЛЬНИЙ ОПИС ПРОГРАМНОГО ПРОДУКТА

Програмний продукт призначений для моделювання роботи статичної нейронної мережі. Модельована мережа може мати необмежену кількість шарів нейронів та необмежену кількість нейронів у кожному шарі.

Кожен шар включає нейрони тільки одного типу. Типи нейронів різних шарів можуть бути різними. У програмне забезпечення закладено можливість зміни типу функції активації нейронів кожного шару. Вбудовані функції активації нейронів: лінійна функція активації; логарифмічна функція активації; функція активації гіперболічний тангенс.

Вихідною інформацією для створення нейронної мережі є її параметри, які знаходяться в файли конфігурації формату ini та можуть бути змінені у будь-якому текстовому редакторі. При цьому відсутня необхідність перебудовувати кінцеве програмне забезпечення. Можливо проводити реініціалізацію мережі під час роботи основної програми.

Програмне забезпечення написано з використанням мови програмування C++ (версія Borland Developer Studio 2006) і може бути використано в програмних продуктах, сумісних з цією мовою.

## 2. ОПИС файлу даних для ініціалізації нейроної мережі

Вихідною інформацією для створення нейронної мережі є її параметри, які знаходяться в файлі конфігурації формату ini, Ці данні можуть бути змінені у будь-якому текстовому редакторі.

Опис нейронної мережі включає такі секції

### Секція опису нейроної мережі в цілому [NN]

Параметри секції

NumInputs – кількість входів мережі

NumLayers – кількість шарів мережі

### Секція опису масштабів виходу в мережі [KY]

Параметри секції

kyXX – значення масштабу на який ділиться вихід XX-го нейрона вихідного шару. XX – номер виходу. Нумерація починається з 1. Кількість параметрів у секції залежить від кількості нейронів у вихідному шарі мережі. У разі, як що цей параметр дорівнює 1, його опис в секції може бути відсутній

### Секція опису довільного шару нейроної мережі [layerUY]

UY – номер шара мережі. Вхідний шар має номер 0 (секція [layer0]). Кількість секцій визначається кількістю шарів у мережі.

Параметри секції

NumNeuron – кількість нейронів у шарі

NeuronAct – номер функції активації нейронів шару: 0 –логарифмічна, 1 – гіперболічний тангенс, 2 – лінійна

NZZ – текстова строчка, в якій перелічені значення засувів та ваги входів нейрону з номером ZZ. Кількість таких параметрів визначається кількістю нейронів у шарі. На початку строки записано розділені знаками «,» (кома) ваги входів нейрона, в кінці строчки записується зсув нейрона. Кількість параметрів у строчці визначається кількістю входів шару мережі (і, відповідно, кількістю входів його нейронів).

## 3. ОПИС КЛАСІВ, З ЯКИХ СКЛАДАЄТЬСЯ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

TNeuralNet – клас опису статичної нейроної мережі. Цей клас є основним класом програмного забезпечення. Опис членів класу та його функцій наведено у табл.1-4

Таблиця 1 – Private члени класу TNeuralNet

Тип	Ідентифікатор	Опис
int	NumLayers	Кількість шарів нейроної мережі
DynamicArray<TNeuronLayer>	vecLayers	Динамічний вектор, який включає шари нейроної мережі
DynamicArray<double>	vecInput	Динамічний вектор, який включає входи нейроної мережі

Таблиця 2 – Private члени - функції класу TNeuralNet

Значення, що повертається	Функція	Опис
---------------------------	---------	------



Значення, що повертається	Функція	Опис
void	Update()	Ініціалізація основних параметрів всіх шарів мережі
void	FinUpdate()	Кінцева ініціалізація всіх параметрів всіх нейронів всіх шарів мережі

Таблиця 3 – Public члени класу TNeuralNet

Тип	Ідентифікатор	Опис
DynamicArray<double>	vecOutput	Вектор виходів нейронної мережі
DynamicArray<double>	vecKY	Вектор масштабів виходів нейронної мережі
int	NumInputs	Кількість входів нейронної мережі
int	NumOutputs	Кількість виходів нейронної мережі

Таблиця 4 – Public члени - функції класу TNeuralNet

Значення, що повертається	Функція	Опис
	TNeuralNet()	конструктор класу
	~TNeuralNet()	деструктор класу
void	LoadIni(String FileName)	Функція завантаження параметрів всіх нейронів всіх шарів нейронної мережі з файлу конфігурації. FileName – шлях та ім'я файла ініціалізації мережі
void	Sim(double Input[])	Функція виклику нейронної мережі на виконання

TNeuronLayer – клас опису окремого шару нейронів у мережі

Опис членів класу та його функцій наведено у табл.5-6

Таблиця 3 – Public члени класу TNeuralNet

Тип	Ідентифікатор	Опис
-----	---------------	------

Тип	Ідентифікатор	Опис
int	NumNeuron	Кількість нейронів у шарі
int	NumInputsPerNeuron	Кількість входів шару
int	NeuronAct	Функція активації нейронів шаров
DynamicArray<TNeuron>	vecNeurons	Динамічний вектор нейронів шару
DynamicArray<double>	vecOutput	Динамічний вектор виходів шару
double	*vecInput	Вказівник на вектор входів шару

Таблиця 4 – Public члени - функції класу TNeuralNet

Значення, що повертається	Функція	Опис
	~TNeuronLayer()	Деструктор класу
void	SetNumNeuron(int NumNeuron)	Функція, що створює нейрони поточного шару
void	LoadIni(TIniFile *IniFile, int iLayer)	Функція завантаження параметрів окремого шару нейронів з файлу конфігурації
void	LoadNeuronsIni(TIniFile *IniFile, int iLayer)	Функція, яка зчитує параметри нейронів у поточному шарі
void	Update()	Функція, яка ініціалізує всі параметри всіх нейронів поточного шару
double	(*ActivationFunction) (double input)	Вказівник на функцію активації нейронів шару
void	Sim()	Функція обчислення виходів всіх нейронів поточного шару

TNeuron – клас опису окремого нейрону

Опис членів класу та його функцій наведено у табл.7-10

Таблиця 7 – Private члени класу TNeuron

Тип	Ідентифікатор	Опис
DynamicArray<double>	vecWeight	Динамічний вектор ваг входів нейрона

Таблиця 8 – Private члени - функції класу TNeuron

Значення, що повертається	Функція	Опис
double	(*ActivationFunction) (double input)	Вказівник на функцію активації нейрона

Таблиця 9 – Public члени класу TNeuron

Тип	Ідентифікатор	Опис
double	*vecInput	Вказівник на вектор входів нейрона
int	NumInputs	Кількість входів нейрона
int	ActFun	Номер функції активації нейрона

Таблиця 10 – Public члени - функції класу TNeuron

Значення, що повертається	Функція	Опис
	~TNeuron()	Деструктор класу
void	LoadIni(TIniFile *IniFile,int iLayer ,int ne)	функція завантаження параметрів нейрона з файлу конфігурації
double	Output()	Функція, що обчислює вихід нейрона за його входами та з урахуванням функції активації
void	SetNumInputs( int NumInputs)	функція, що встановлює кількість входів нейрона
void	SetActFun(int iActFun)	Функція вибору функції активації нейрона

Локальні функції, які не входять до класів

Опис функцій наведено у табл.11

Таблиця 10 – Локальні функції, які не входять до класів

Значення, що повертається	Функція	Опис
double	LogSig(double input)	inline логарифмічна функція активації нейрона
double	TanSig(double input)	inline функція активації нейрона гіперболічний тангенс
double	PureLin(double input)	inline лінійна функція активації нейрона
String	DivideString(String &text,char div,bool bRetRest)	Розділяє текстову строку text на дві частини по символу div і повертає першу частину
int	StringToFloatVec(String tx,double *v)	допоміжна функція перетворення строки, що складається з набору чисел типу double, розділених знаками «кома»/«,», у вектор типу double

## 4. ПРИКЛАД РОБОТИ З ПРОГРАМНИМ ПРОДУКТОМ

Розглядається процес створення, ініціалізації та використання двошарової статичної нейронної мережі, яка має 2 нейрони у першому шарі та один нейрон у вихідному шарі. Кількість входів мережі – 3. Функція активації першого шару – гіперболічний тангенс, другого – лінійна.

### 4.1. ПРИКЛАД ФАЙЛА ІНІЦІАЛІЗАЦІЇ

Файл ініціалізації (ім'я NN.INI) має вигляд

Секція опису нейронної мережі в цілому

[NN]

NumInputs=3 // Кількість входів

NumLayers=2 // Кількість шарів

Секція опису масштабів виходу в мережі

[KY]

ky1= 1 // Масштаб/вага виходу першого нейрона вихідного шару.

Секція опису вхідного шару нейронної мережі

[layer0]

NumNeuron=2 // Кількість нейронів у шарі. 0 –логарифмічна, 1 – гіперболічний тангенс, 2 – лінійна

NeuronAct=1 // Номер функції активації нейронів шару

// Засув та ваги входів першого нейрону

N0= 1, 2, 3, 4

// Засув та ваги входів другого нейрону

N1= 4, 3, 2, 1

Секція опису вихідного шару нейронної мережі

[layer1]

NumNeuron=1 //Кількість нейронів у шарі

NeuronAct=2 // Номер функції активації нейронів шару. 0 –логарифмічна, 1 – гіперболічний тангенс, 2 – лінійна

// Засув та ваги входів другого нейрону

N0= 1, 1, 2

### 4.2. ПРИКЛАД СТВОРЕННЯ, ІНІЦІАЛІЗАЦІЇ ТА ВИКОРИСТАННЯ СТАТИЧНОЇ НЕЙРОННОЇ МЕРЕЖІ

```
// Створення нейронної мережі як об'єкту класу TNeuralNet
```

```
TNeuralNet DiagnNN;
```

```
//Зчитування з файлу конфігурації NN.INI (знаходиться у поточній папці) параметрів нейронної мережі, її шарів та нейронів у всіх шарах
```

```
DiagnNN.LoadIni("NN.INI");
```

```
//створення вектору входів мережі
```

```
double input[3];
```

```
// Ініціалізація входив мережі (у прикладі наведено довільні значення входів мережі)
```

```
input[0]=0,2;  
input[1]=0,5;  
input[2]= -0,1;  
// Запуск нейронної мережі на розрахунок  
DiagnNN.Sim(input);  
// Результат розрахунку  
DiagnNN.vecOutput[0];
```