

*Босых О.С.,  
студент магистратуры 2 курса  
Хакасский государственный университет им. Н.Ф. Катанова  
Россия, г. Абакан  
Научный руководитель: Голубничий А.А.*

**ПОСТРОЕНИЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ  
ПОСРЕДСТВОМ NEURALNET НА ЯЗЫКЕ R  
(НА ПРИМЕРЕ НАБОРА ДАННЫХ IRIS)**

***Аннотация:** В статье описывается процесс построения простейшей нейронной сети в пакете neuralnet. Рассматривается алгоритм подготовки данных на примере набора данных iris. Описывается процесс обучения нейронной сети. Оценивается качество классификации данных полученной нейронной сетью.*

***Ключевые слова:** R, искусственные нейронные сети, neuralnet.*

***Abstract:** The article describes the process of building the simplest neural network in the neuralnet package. The algorithm of data preparation on the example of the iris data set is considered. Describes the process of learning a neural network. The quality of data classification obtained by the neural network is evaluated.*

***Key words:** R, artificial neural networks, neuralnet.*

Процесс создания искусственных нейронных сетей на разных языках программирования требует различных усилий. Реализация алгоритма обучения нейронной сети, без использования специализированных модулей, одинаково сложная задача для большинства языков программирования. В целях упрощения, в рамках данного исследования, будет использован язык программирования R [1] и пакет neuralnet [2]. Исходными данными для обучения выступит

встроенный в базовую версию языка набор данных `iris`, содержащий сведения о морфометрии разных видов ириса [3].

Прежде чем приступить к созданию нейронной сети, необходимо выгрузить исходные данные и подготовить их к обучению (рис. 1.).

```
1 #Построение нейронной сети с помощью пакета R neuralnet
2
3 #Загрузка данных (отбор данных по двум сортам)
4 df <- iris[1:100,]
5
6 #Загрузка данных и разбиение на тренировочный (70%) и тестовый (30%) наборы:
7 df$Species <- ifelse(df$Species=="setosa", 1, 0)
8 ind <- sample(2, nrow(df), replace = TRUE, prob = c(0.7, 0.3))
9 trainset = df[ind == 1,]
10 testset = df[ind == 2,]
11 trainset[,1:4] <- as.data.frame(scale(trainset[,1:4]))
12 testset[,1:4] <- as.data.frame(scale(testset[,1:4]))
```

### *Рисунок 1. Исходный код загрузки и обработки данных*

Для упрощения работы с данными уменьшим исходный набор со 150 до 100 значений, тем самым уберем из выборки один из видов ириса (останутся виды `setosa` и `versicolor`). Еще одним методом упрощения будет передача численного значения соответствующим видам, так, заменим вид `setosa` значением 1, а `versicolor` – 0, данную операцию проведем тернарным оператором `ifelse`. Используя функцию `sample`, случайным образом разобьем исходный набор данных на тренировочный (70%) и тестовый (30%). Для работы с нейронной сетью необходимо нормализовать данные, эту процедуру можно выполнить как вручную, используя функции `min`, `max` и соответствующие арифметические операции, так и используя стандартную функцию `scale`.

Как уже было сказано, обучение нейронной сети происходит благодаря пакету `neuralnet`. Данный пакет размещен в официальной репозитории CRAN, поэтому его можно загрузить и подключить стандартным набором функций `install.packages()` и `library()` (рис. 2.). Сам процесс тренировки сети осуществляется через функцию, аналогичную названию пакета, путем передачи в формате формулы выходного параметра и входных параметров, набора для обучения и структуры самой сети.

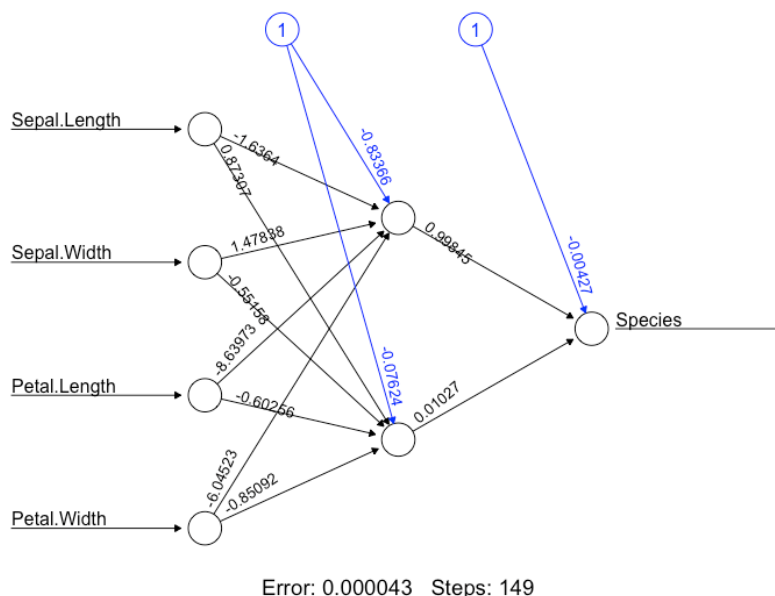
```

14 #Установка и загрузка пакета neuralnet:
15 install.packages("neuralnet")
16 library(neuralnet)
17
18 #Тренировка нейронной сети с помощью функции neuralnet
19 network <- neuralnet(Species ~ Sepal.Length + Sepal.Width + Petal.Length
20                       + Petal.Width, trainset, hidden = 2)
21 network
22 network$result.matrix
23 plot(network)

```

**Рисунок 2. Исходный код установки и загрузки пакета neuralnet и тренировки нейронной сети**

Определение вида растения по морфометрии является естественной задачей, по этой причине нет необходимости выстраивать сложную структуру нейронной сети. В нашем примере мы ограничиваемся нейронной сетью с одним скрытым слоем с двумя нейронами (рис. 3.).



**Рисунок 3. Нейронная сеть**

Тестирование качества нейронной сети будем проводить, используя заранее подготовленную выборку из исходных данных (рис. 4.). С помощью команды subset подготовим временный набор данных с исключением переменной Species. Используя функцию compute, проведем расчет исключенной нами переменной с использованием нейронной сети. Далее, округлив значения до целых, построим таблицу сравнения исходных данных (actual) и предсказанных с помощью нейронной сети (prediction) (рис. 5.).

```

25 #Тестирование результатов работы нейронной сети
26 temp_test <- subset(testset, select = c("Sepal.Length", "Sepal.Width",
27                                       "Petal.Length", "Petal.Width"))
28 head(temp_test)
29 network.results <- compute(network, temp_test)
30 results <- data.frame(actual = testset$Species,
31                       prediction = network.results$net.result)
32
33 roundedresults <- data.frame(sapply(results, round, digits=0))
34 table(actual, prediction)

```

*Рисунок 4. Исходный код тестирования нейронной сети*

```

> table(actual, prediction)
      prediction
actual 0  1
      0 13  0
      1  0 20

```

*Рисунок 5. Оценка работы нейронной сети*

Качество работы нейронной сети можно оценить по количеству значений, лежащих вне диагонали – чем меньше значений, тем точнее работа нейронной сети. В нашем примере значений, не относящихся к диагонали, обнаружено не было, что свидетельствует о точности решения исходной задачи.

### ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ

1. R: The R Project for Statistical Computing [Электронный ресурс] URL: <https://www.r-project.org> (дата обращения: 12.01.2019).
2. CRAN – Package neuralnet [Электронный ресурс] URL: <https://cran.r-project.org/web/packages/neuralnet/index.html> (дата обращения: 12.01.2019).
3. Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) The New S Language. Wadsworth & Brooks/Cole. (has iris3 as iris.)