**1.1.1. Биологические нейронные сети: основы**

Для начала рассмотрим биологические нейронные сети, явившиеся прообразом искусственных нейронных сетей (ИНС). Биологические нейронные сети состоят из биологических нейронов, которые имеют множество взаимных связей.

Биологический нейрон – функциональная единица нервной системы, обладающая специфическими проявлениями возбудимости.

Нейроны бывают двух типов: моторные нейроны (мотонейроны) и нейроны-рецепторы, реагирующие на внешние условия.

Мотонейроны делятся на два типа: тормозные и возбуждающие. В отличие от тормозных мотонейронов, возбуждающие мотонейроны понижают мембранный потенциал клетки-адресата. Мембранный потенциал - это характеристика клетки, которая определяет возможность клетки по передаче сигнала.

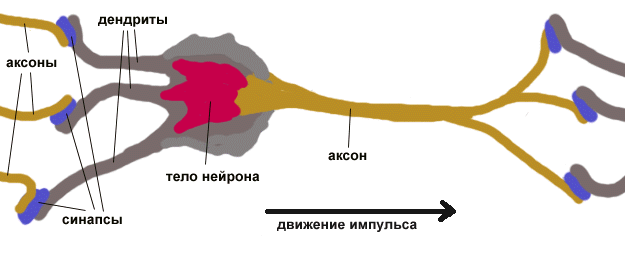


Рис 1. Схематическое изображение нейрона и его роли в передаче сигнала в сети

На рисунке 1 схематически изображен нейрон. Он состоит из аксона, синапсов, дендритов и тела.

Аксон это длинный отросток нейрона, который предназначен для передачи сигнала далее к другому нейрону или клетке.

Дендрит – отросток нейрона, который гораздо короче аксона, и предназначен для приема сигнала. Дендрит более разветвлен, чем аксон. К дендритам присоединены все аксоны других нейронов. У нейрона может быть множество дендритов и в среднем только 1 аксон.

Синапс – кусочек мембраны на аксоне и клетке-адресате, между которыми передается электрохимический импульс. Именно благодаря синапсам мы можем измерять силу связи между клетками, а при необходимости и менять.

Итак, биологическая нейронная сеть состоит из множества простых взаимосвязанных нейронных клеток, способна передавать сигнал, и может выполнять сложные задачи.

# 1.1.2. Принципы обучения биологической нейронной сети

Нейронные сети есть практически во всех живых организмах. К сожалению, человечеству еще очень мало известно о том, как работают разные системы памяти. Пока ученые только моделируют сети, которые могли бы себя вести, как нейронные сети в живых организмах.

У нейронных сетей много важных свойств, но ключевое из них – это способность к обучению. Обучение нейронной сети в первую очередь заключается в изменении «силы» синапсических связей между нейронами.

# 1.2. Искусственные нейронные сети

В наши дни возрастает необходимость в системах, которые способны не только выполнять однажды запрограммированную последовательность действий над заранее определенными данными, но и способны сами анализировать вновь поступающую информацию. Здесь хорошо себя зарекомендовали искусственные *нейронные сети* – самообучающиеся системы, имитирующие деятельность человеческого мозга. Интеллектуальные системы, основанные на использовании искусственных нейронных сетей (ИНС), позволяют легко решать различные сложные проблемы.

Искусственная нейронная сеть – это значительным образом упрощенная модель биологической нейронной сети.

Сейчас развиты электронные технологии, позволяющие создать систему, действующую по принципу функционирования нейронных сетей, на основе электронных элементов. Другими словам мы можем моделировать нейронную сеть с помощью электронных элементов. Работа нейронной сети сводится к формированию некоторого полезного выходного набора данных в ответ на совокупность входных данных, т.е. реализации некоторого функционального преобразования данных. Самым важным свойством биологических нейронных сетей является их способность обучаться и в результате обучения повышать свою производительность. Такой способностью обладают и искусственные НС.

Обучение нейронной сети происходит при изменении синапсических связей между клетками. Сеть обучается, чтобы для некоторого множества входов давать желаемое множество выходов. Далее описаны результаты работ по созданию искусственной нейронной сети с помощью электронного конструктора Arduino.

Arduino – это электронный конструктор, инструмент для создания электронных устройств, аппаратная вычислительная платформа. Arduino применяется для создания электронных устройств с возможностью приема сигналов от различных цифровых и аналоговых датчиков, которые могут быть подключены к нему, и управления различными исполнительными устройствами. Устройства, основанные на Arduino, могут работать самостоятельно или взаимодействовать с программным обеспечением на компьютере (например, Flash, Processing, MaxMSP).

Далее приведена программа, по которой работает моя нейронная сеть:

///int m[]={0,0,0};

int m[2];

int arr[]={1,2,3};

int switchPin1 =6; ///knopki

int switchPin2 =7;

int sensePin1 =0; ///sensori

int sensePin2 =1;

int sensePin3 =2;

int ledPin1 = 13; ///pokazivaet kogda nazimaem fotorezistor

int ledPin2 = 12; ///pokazivaet kogda nazimaem fotorezistor

int ledPin3 = 8; ///pokazivaet kogda nazimaem fotorezistor

int ledPin4 = 2; ////knopka obuchenia + migaet kogda pravilno(pri rabote)

int ledPin5 = 3; ////idet rabota

int ledPin6 = 4; ///migaet kogda zakon4il obuchenie

int ledPin7 = 9; ///migaet vo vremya obuchenia

int a = 0; ///peremennaya dlia funkzii lampslight

int b = 0; ///peremennaya dlia funkzii lampslight

int c = 0; ///peremennaya dlia funkzii lampslight

int sum = 0; ///summator(ispolzuetsya pri rabote)

int numb=0;

boolean lastButton = LOW;

boolean ledOn = false;

boolean currentButton = LOW;

void setup()

{

analogReference(DEFAULT);

pinMode(ledPin1, OUTPUT);

pinMode(ledPin2, OUTPUT);

pinMode(ledPin3, OUTPUT);

pinMode(ledPin4, OUTPUT);

pinMode(ledPin5, OUTPUT);

pinMode(ledPin6, OUTPUT);

pinMode(ledPin7, OUTPUT);

pinMode(switchPin1, INPUT);

pinMode(switchPin2, INPUT);

Serial.begin(9600);

}

boolean debounce(boolean last)

{

boolean current = digitalRead(switchPin1);

if (last != current)

{

delay(5);

current = digitalRead(switchPin1);

}

return current;

}

void lampslight() //// naprotiv kazdogo sensora zazigaetsya lampa pri nazatii

{

a = 0;

b = 0;

c = 0;

int val1 = analogRead(sensePin1);

int val2 = analogRead(sensePin2);

int val3 = analogRead(sensePin3);

if (val1 < 30) {a=1; digitalWrite(ledPin1, HIGH); }

if (val1 > 30) { digitalWrite(ledPin1, LOW); }

if (val2 < 30) {b=1; digitalWrite(ledPin2, HIGH);}

if (val2 > 30) { digitalWrite(ledPin2, LOW);}

if (val3 < 30) {c=1; digitalWrite(ledPin3, HIGH);}

if (val3 > 30) { digitalWrite(ledPin3, LOW);}

}

void rabota() ///poka ne nazata knopka obuchenia proishodit rabota

{

digitalWrite(ledPin5,HIGH);

///rabota

Serial.println("WORK STARTED");

lampslight();

numb=a+b+c;

Serial.print("numb= ");

Serial.println(numb);

for (int v=0; v<3; v++)

{Serial.print("m=");

Serial.println(m[v]);}

/// Serial.println(m[0]);

/// Serial.println(m[1]);

/// Serial.println(m[2]);

sum=0;

for (int j=0; j<numb; j++){sum=sum+m[j];}

Serial.print("sum=");

Serial.println(sum);

delay(1000);

if (sum>0)

{

digitalWrite(ledPin5,LOW);

digitalWrite(ledPin4,HIGH);

delay(3000);

digitalWrite(ledPin4,LOW);

digitalWrite(ledPin5,HIGH);

}

}

void obuchenie()

{

m[0]=0;

m[1]=0;

m[2]=0;

Serial.println("Initial assignment of m array");

for (int i=0; i<3; i++) //// for 3 attempts

{

lampslight();

numb=a+b+c; //// число зажатых фотoрезисторов

for (int j=1; j<=3; j++) //// j - номер сумматора

{

if (j>numb) {m[j-1]=m[j-1]-1;}

if (j == numb) {m[j-1]=m[j-1]+1;}

}

Serial.println("Learning");

Serial.println(i);

digitalWrite(ledPin7,HIGH); ////zapomnil kombinatsiyu

delay(1000);

digitalWrite(ledPin7,LOW);

delay(500);

}

digitalWrite(ledPin6,HIGH); ////obuchenie okon4eno

delay(3000);

digitalWrite(ledPin6,LOW);

for (int v=0; v<3; v++)

{Serial.print("m=");

Serial.println(m[v]);}

}

void loop()

{

currentButton = debounce(lastButton);

if (lastButton == LOW && currentButton == HIGH){ledOn = !ledOn;}

lastButton = currentButton;

digitalWrite(ledPin4, ledOn);

lampslight();

Serial.println("Lamps tested ");

if (digitalRead(ledPin4) == HIGH) ////( esli gorit lampa 4, то включаем лампу 5, обнуляем ячейки и pereklu4aem rezim v obuchenie

{

delay(500);

digitalWrite(ledPin5,LOW); ///Turn led5 on и обнуляем все ячейки массива m

if (digitalRead(switchPin2) == HIGH) /// Если нажата кнопка 2, то обучение

{

Serial.println("Learning from loop");

obuchenie();

}

}

if (digitalRead(ledPin4) == LOW) //// esli не gorit lampa 4, то работа

{

Serial.println("Work from loop");

rabota();

}

}