ФГАОУ ВО «Севастопольский государственный университет»

Институт информационных технологий   
и управления в технических системах

ОТЧЁТ  
ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

«Исследование однослойного персептрона»

по дисциплине «Нейрокомпьютерные технологии»

Выполнила:  
студентка группы ИС/м-21-1-з  
Ускова Екатерина Дмитриевна

Севастополь

2022

# Цель работы

Углубление теоретических знаний в области архитектуры нейронных сетей с пороговыми активационными функциями, исследование свойств однослойного персептрона и правила его обучения, приобретение практических навыков обучения и моделирования однослойной сети при решении простых задач классификации.

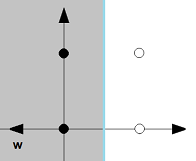
# Ход работы

Для заданной матрицы входных данных P и заданного вектора выходных значений T (вариант 4), разработан простой персептрон, решающий задачу классификации 2-х классов. Решение найдено с помощью графического построения границы решения и вычисления весов и смещения вручную. Полученное решение протестировано с помощью компьютера для всех входных векторов (столбцов матрицы P).





Решение, найденное вручную:



Вектор весов, ортогональный границе w = [-1 0]T. Произвольная точка на границе p = [0.5 0]T. wTp = -0.5 + b = 0 => b = 0.5.

Тестирование с помощью компьютера:

P = [0 1 1 0; 0 0 1 1]

T = [1 0 0 1]

W = [-1 0]

b = 0.5

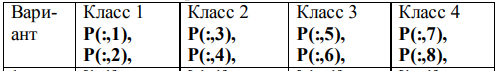
a = ann\_hardlim\_activ(W \* P + b)

disp(a)

printf("Количество ошибок: %d\n", sum(a - T))

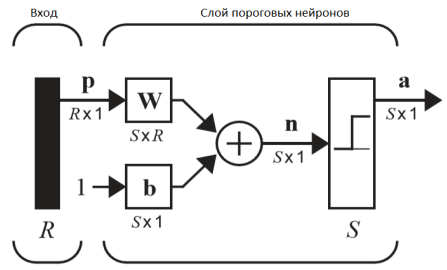


Даны четыре класса, каждый из которых представлен 2-мя точками (столбцами матрицы P), указанными в таблице.

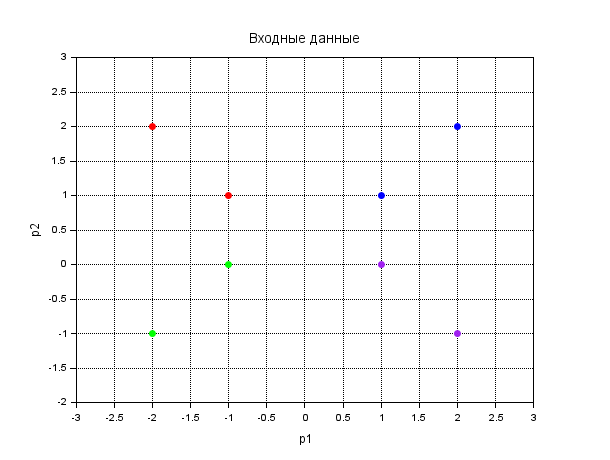




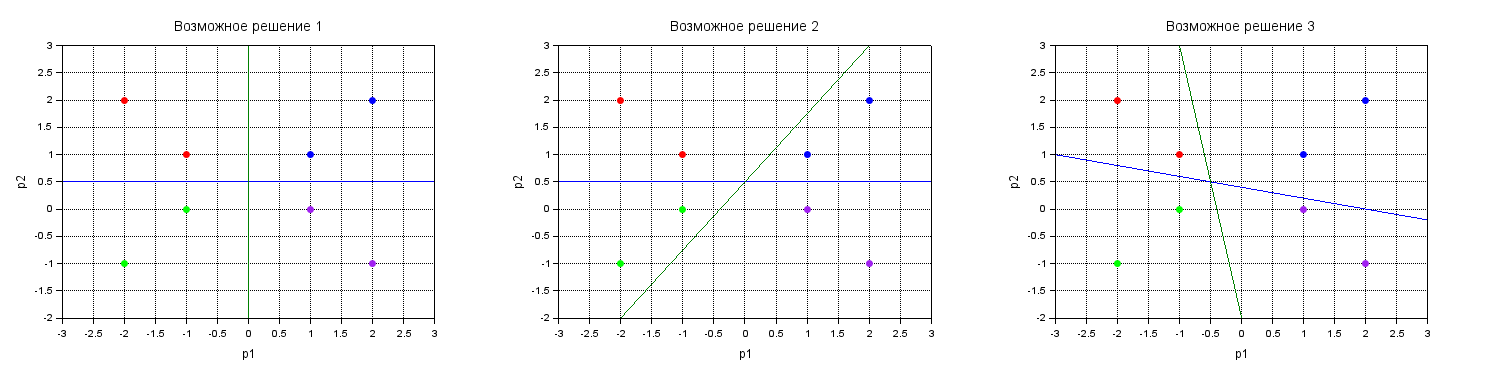
Разработана структурная схема персептрона, распознающего эти 4 класса. Персептрон, распознающий данные классы, будет иметь стандартную схему, где число нейронов S = 2.



Выполнен предварительный анализ задачи, изображены все точки четырех классов и построены графически возможные границы решений персептрона. Тточки, относящиеся к разным классам, отображены разными цветами.



Данная задача имеет бесконечное множество решений, некоторые из них:



Анализируя границы решений, заданы допустимые целевые значения выходов персептрона (определена матрица T) для всех входных точек, представленных столбцами матрицы данных P.

T = [1 1 0 0 0 0 1 1;

1 1 1 1 0 0 0 0]'

Здесь вектор-столбец [1; 1] соответствует первому классу, [0; 1] – второму, [0; 0] – третьему и [1; 0] – четвертому.

Изучена функция ann\_PERCEPTRON и, используя её, обучен персептрон правильному распознаванию входных классов.

P = [1 2 -1 -2 -1 -2 1 2;

1 2 1 2 0 -1 0 -1]

T = [1 1 0 0 0 0 1 1;

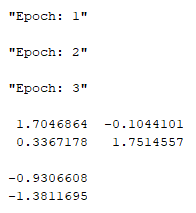
1 1 1 1 0 0 0 0]

*// обучение персептрона*

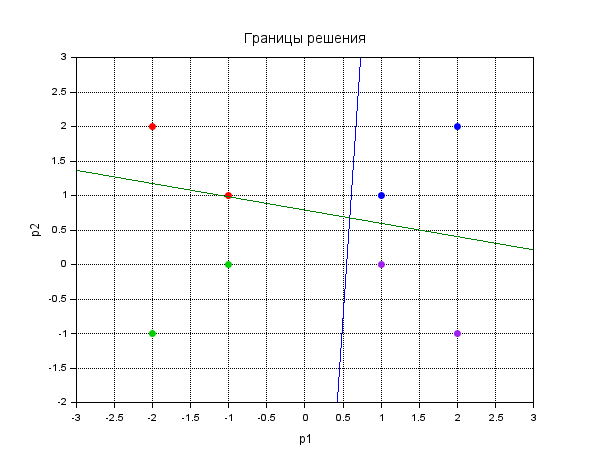
[w, b] = ann\_PERCEPTRON(P, T);

disp(w)

disp(b)



Границы полученного решения:



Написана программа, которая:

* отображает диаграмму размещения входных точек из P на плоскости с координатами (p1, p2);
* обучает персептрон;
* накладывает на диаграмму входных точек границы решения после обучения персептрона;
* выполняет тестирование полученного решения для всех заданных входных данных, а также для дополнительно выбранных точек из областей принадлежности входных классов.)

Сгенерирован тестовый набор случайных точек. Точки из обучающего и тестового наборов классифицированы ранее обученным персептроном. Тестовые точки отображены незакрашенными кругами, точки из обучающего набора – закрашенными, результаты классификации изображены с помощью использования различных цветов.

function **c**=get\_class\_color(**A**)

*// функция для соотнесения цвета классу*

if **A** == [1; 1] then

**c** = "blue"

elseif **A** == [0; 1] then

**c** = "red"

elseif **A** == [0; 0] then

**c** = "scilabgreen2"

elseif **A** == [1; 0] then

**c** = "purple"

else

**c** = "black"

end

endfunction

*// запуск сети на обучающем наборе*

res = ann\_PERCEPTRON\_run(P, w, b);

disp(res)

clf

*// отрисовка обучающих точек*

for j=1:size(P,2)

scatter(P(1,j), P(2,j), , get\_class\_color(res(:, j)), 0)

end

*// тестовые точки*

test = [6 \* rand(1, 5) - 3; 5 \* rand(1, 5) - 2]

disp(test)

*// запуск сети на тестовом наборе*

res\_test = ann\_PERCEPTRON\_run(test, w, b);

*// отрисовка тестовых точек*

for j=1:size(test,2)

scatter(test(1,j), test(2,j), , get\_class\_color(res\_test(:, j)))

end

xtitle("Классификация", "p1", "p2")

zoom\_rect([-3 -2 3 3])

xgrid

*// границы решения*

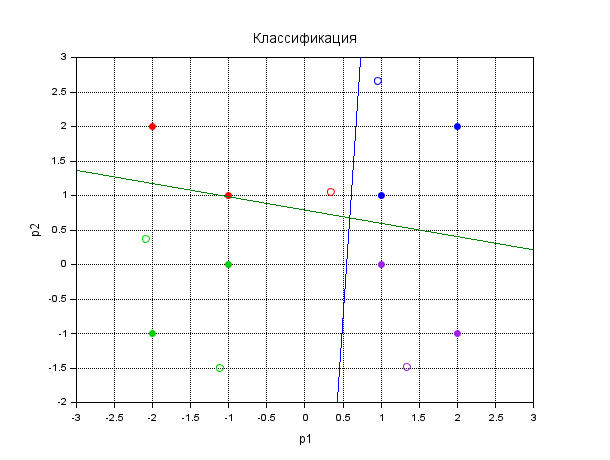
x = (-3:0.1:3)

y1 = (-b(1)-w(1, 1)\*x)/w(1, 2)

y2 = (-b(2)-w(2, 1)\*x)/w(2, 2)

plot(x, y1, x, y2)

Результат классификации:



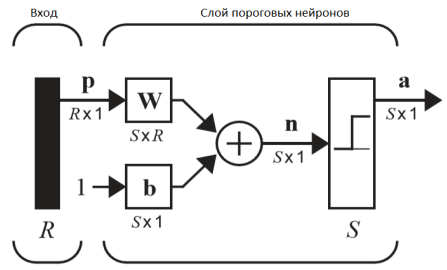
Таким образом, классификация точек соответствует полученным границам решения.

# Вывод

Выполняя эту работу, я углубила теоретические знания в области архитектуры нейронных сетей с пороговыми активационными функциями, исследовала свойства однослойного персептрона и правила его обучения, приобрела практические навыки обучения и моделирования однослойной сети при решении простых задач классификации.

# Ответы на контрольные вопросы:

6.1 *Нарисуйте схему однослойного* *персептрона, объясните все обозначения, запишите формулу вычисления вектора выходных значений и объясните её.*



p – входной вектор

R – длина входного вектора p

W – матрица весов

b – вектор смещений

S – количество нейронов в слое

n = W\*p + b – где-то было написано как это называется

a = hardlim(n) – выход функции активации персептрона

Входом n функции активации персептрона служит сумма смещения b и произведения W\*p. Эта сумма преобразуется функцией активации f, на выходе которой формируется выход персептрона a.

6.2 *Сколько линейно-разделимых классов способен распознать персептрон, содержащий S нейронов? Объясните почему?*

2S. Поскольку каждый нейрон способен разделить пространство решений на два класса.

6.3. *Изобразите простой персептрон. Запишите уравнение границы решения.*

y = (-b - w(1) \* x) / w(2)

6.4. *Как построить границу решения простого персептрона, если известны его веса и смещения? Как определить область входного пространства, где выход персептрона будет равен 1?*

По значениям весов и смещений вывести уравнение прямой.

Выход персептрона равен 1 в той области, в которую направлен вектор весов.

6.5. *Как можно графически построить границу решения персептрона. Приведите пример построения.*

В коде

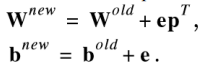
6.6. *Как связана ориентация вектора весов с границей решения нейрона персептрона?*

Вектор весов направлен перпендикулярно границе решения.

6.7*. Запишите правило обучения персептрона в векторной и матричной форме?*







6.8. *Чему равны значения ошибки персептрона?*



Правильный ответ минус полученный.

6.9. *В каком направлении корректируются значения векторов весов нейронов персептрона?*

Вектор весов отдельного нейрона персептрона в ходе обучения изменяется на величину очередного входного вектора p c учетом ошибки, которая равна -1, 0 или 1.

6.10. *Сформулируйте теорему сходимости персептрона.*

Правило обучения персептрона обеспечивает схождение весов к требуемым значениям, обеспечивающим правильную классификацию, за конечное число шагов при условии существования решения.

6.11. *Какой класс задач распознавания способен решать персептрон?*

Линейно-разделимых (линейно-сепарабельных).

6.12. *Как при известном расположении распознаваемых классов корректно сформировать матрицу целевых значений выходов персептрона? Приведите пример.*

Необходимо отталкиваться от предположительных границ решения. Для каждой из них решить, с какой стороны будет 0, а с какой 1. Пример в работе.

6.13. *Какие встроенные функции модуля Neuralnetwork 2.0 используются для обучения и моделирования персептрона? Приведите примеры их вызова.*

ann\_PERCEPTRON и ann\_PERCEPTRON\_run

[w, b] = ann\_PERCEPTRON(P, T);

res = ann\_PERCEPTRON\_run(P, w, b);

6.14. *Объясните алгоритм, реализуемый функцией ann\_PERCEPTRON.*

Инициализирует веса и смещения случайными числами, проверяет решение, корректирует относительно ошибок пока ошибки не исчезнут.