**Московский авиационный институт**

**(Национальный исследовательский университет)**

Факультет прикладной математики и физики

Кафедра вычислительной математики и программирования

**Лабораторная работа № 1**

по курсу «Нейроинформатика»

Тема: Персептроны. Процедура обучения Розенблатта.

Студент: Жерлыгин М.А.

Группа: 80-408б

Преподаватель: Аносова Н.П.

Оценка:

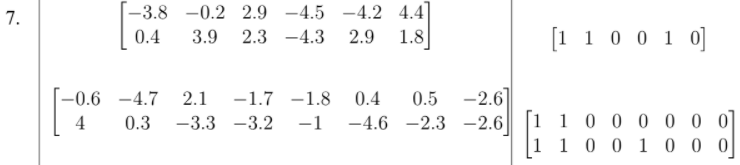
Москва, 2021

**Задание 1**

**1) Постановка задачи:** Исследование свойств персептрона Розенблатта и его применение для решения задачи распознавания образов.

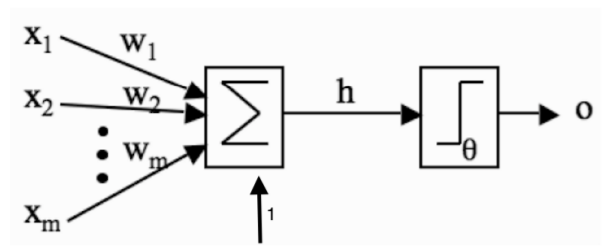
1. Для первой обучающей выборки построить и обучить сеть, которая будет правильно относить точки к двум классам. Отобразить дискриминантную линию и проверить качество обучения.
2. Изменить обучающее множество так, чтобы классы стали линейно неразделимыми. Проверить возможности обучения по правилу Розенблатта.
3. Для второй обучающей выборки построить и обучить сеть, которая будет правильно относить точки к четырем классам. Отобразить дискриминантную линию и проверить качество обучения. Проверить качество, на случайно заданном множестве состоящим из пяти элементов.

Входные данные (вариант №7):



**2) Ход работы:**

Для решения этой задачи необходимо воспользоваться Перцептроном Розенблата, который имеет следующую структуру:



Чтобы реализовать слой таких перцептронов можно воспользоваться представлением весов и смещений перцептронов как матрицу (n+1)×m, где n - число входов, а m - число выходов.

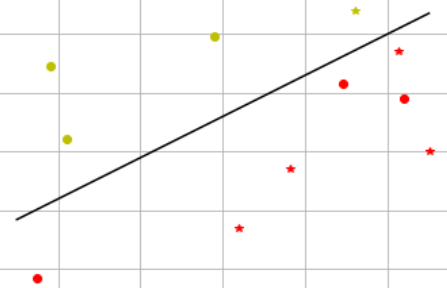
Алгоритм обучения:

def train(df, n\_input, n\_neurouns, eta=0.3, log\_file=None, max\_iter=1e6):  
 if log\_file is not None:  
 f = open(log\_file, 'w')  
 x = df.iloc[:, :n\_input]  
 y = df.iloc[:, n\_input:]  
 w = np.random.uniform(-2, 2, (n\_input, n\_neurouns))  
 idx = 0  
 k\_iter = 0  
 while True:  
 if log\_file is not None:  
 f.write(f'Step {idx}:\nIter {k\_iter}\n{w}\n')  
 if idx == df.shape[0] or k\_iter == max\_iter:  
 break  
 tmp = x.iloc[idx] @ w  
 res = out(tmp)  
 if all(res == y.iloc[idx]):  
 idx += 1  
 else:  
 err = y.iloc[idx] - res  
 if log\_file is not None:  
 f.write(f'Err:\n{err}\n')  
 f.write(f'X:\n{x.iloc[idx]}\n')  
 f.write(f'Y:\n{y.iloc[idx]}\n')  
 f.write(f'Out\n{res}\n')  
 w += np.array([[eta \* i \* j for j in err] for i in x.iloc[idx]])  
 idx = 0  
 k\_iter += 1  
 if log\_file is not None:  
 f.close()  
 return w

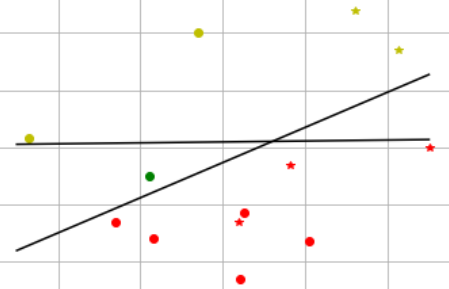
Функция активации:

def out(x):  
 return np.array([1 if i >= 0 else 0 for i in x])

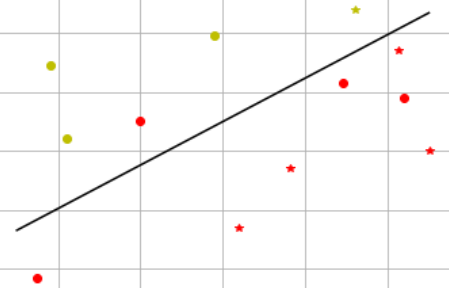
После обучения сети (1 датасет):



После обучения сети (2 датасет):



В качестве линейно неразделимых классов я взял первый датасет и добавил в него одну точку:



**3) Вывод**

Выполнив первую лабораторную работу по курсу «Нейроинформатика», я узнал о перцептроне Розенблата, который имеет историческое значение для всей науки.

Перцептрон Розенблата несмотря на свою простоту и умение классифицировать объекты, имеет ряд очевидных недостатков:

1. Неозможность классификации на неразделяемых данных.
2. Плохая сходимость и неустойчивость.

Эти недостатки решают более современные архитектуры.