**Московский авиационный институт**

**(Национальный исследовательский университет)**

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Нейроинформатика»

**Лабораторная работа № 1**

Тема: Персептроны. Процедура обучения Розенблатта

Студент: Хренов Геннадий

Группа: 80-407Б

Преподаватель: Аносова Н. П.

Дата:

Оценка:

Москва, 2021

1. Цель работы

Исследование свойств персептрона Розенблатта и его применение для решения задачи распознавания образов.

1. Основные этапы работы

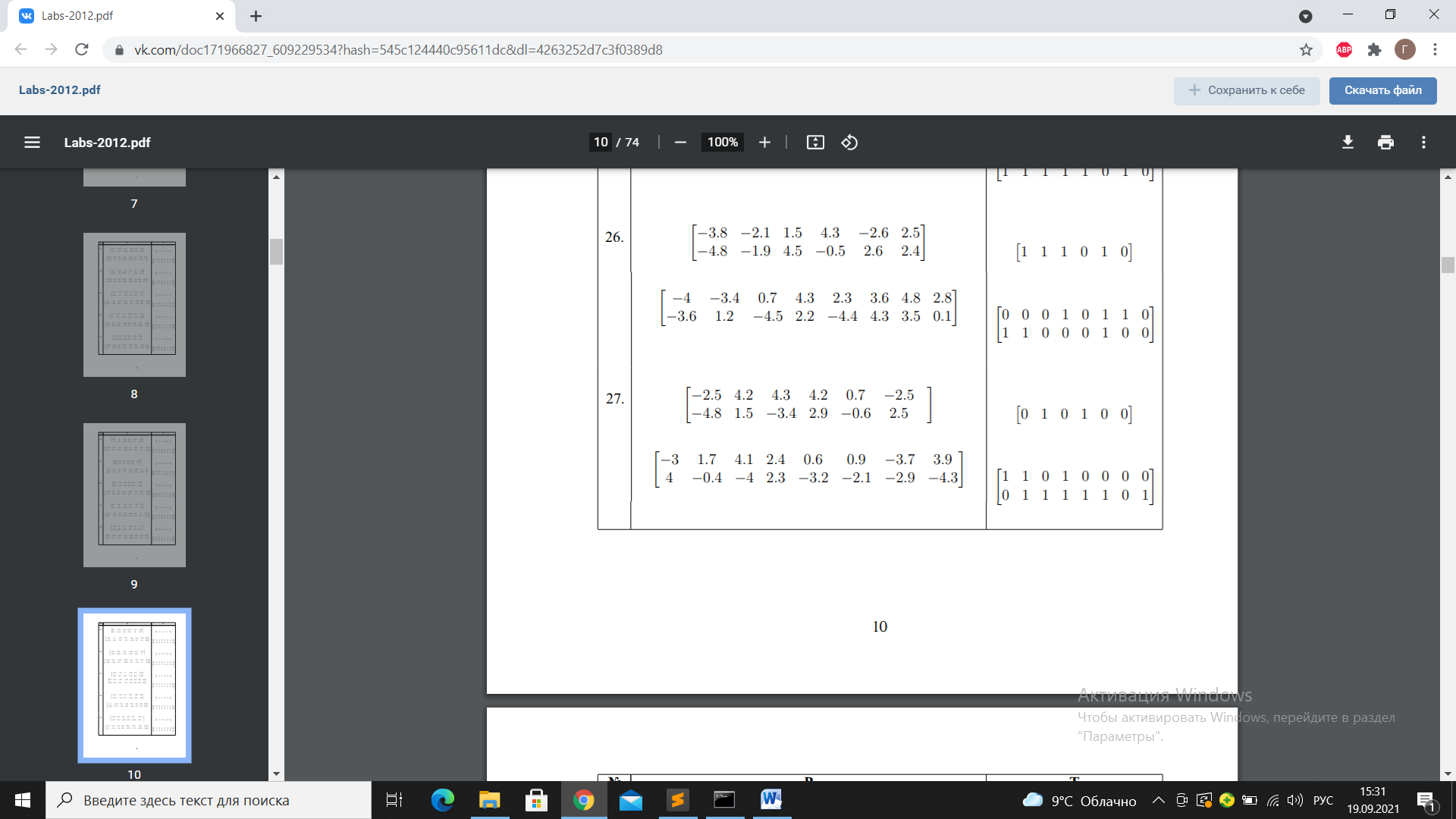
1. Для первой обучающей выборки построить и обучить сеть, которая будет правильно относить точки к двум классам. Отобразить дискриминантную линию и проверить качество обучения.

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.

1. Выполнение работы

3.1 Обучающее множество



3.2 Реализация персептрона

**class** **Perseptron:**

**def** \_\_init\_\_**(**self**,**nin**):**

self**.**W **=** np**.**random**.**uniform**(-**1**,** 1**,** nin**)**

self**.**b **=** np**.**random**.**uniform**(-**1**,** 1**)**

self**.**err **=** 0.

**def** forward**(**self**,** x**):**

**return** np**.**dot**(**x**,** self**.**W**)** **+** self**.**b

**def** update**(**self**,**x**,**lr**=**0.1**):**

self**.**W **+=** lr **\*** self**.**err **\*** x

self**.**b **+=** lr **\*** self**.**err

**def** fit**(**self**,** X**,** Y**):**

flag **=** **True**

count **=** 0

**while** flag**:**

**for** x**,**y **in** **zip(**X**,**Y**):**

self**.**err **=** y **-** **(**0 **if** self**.**forward**(**x**)** **<** 0 **else** 1**)**

**if** **(**self**.**err **==** 0**):**

count**+=**1

**if** count **==** **len(**X**):**

flag **=** **False**

**break**

**else:**

self**.**update**(**x**)**

count **=** 0

**def** get\_param**(**self**):**

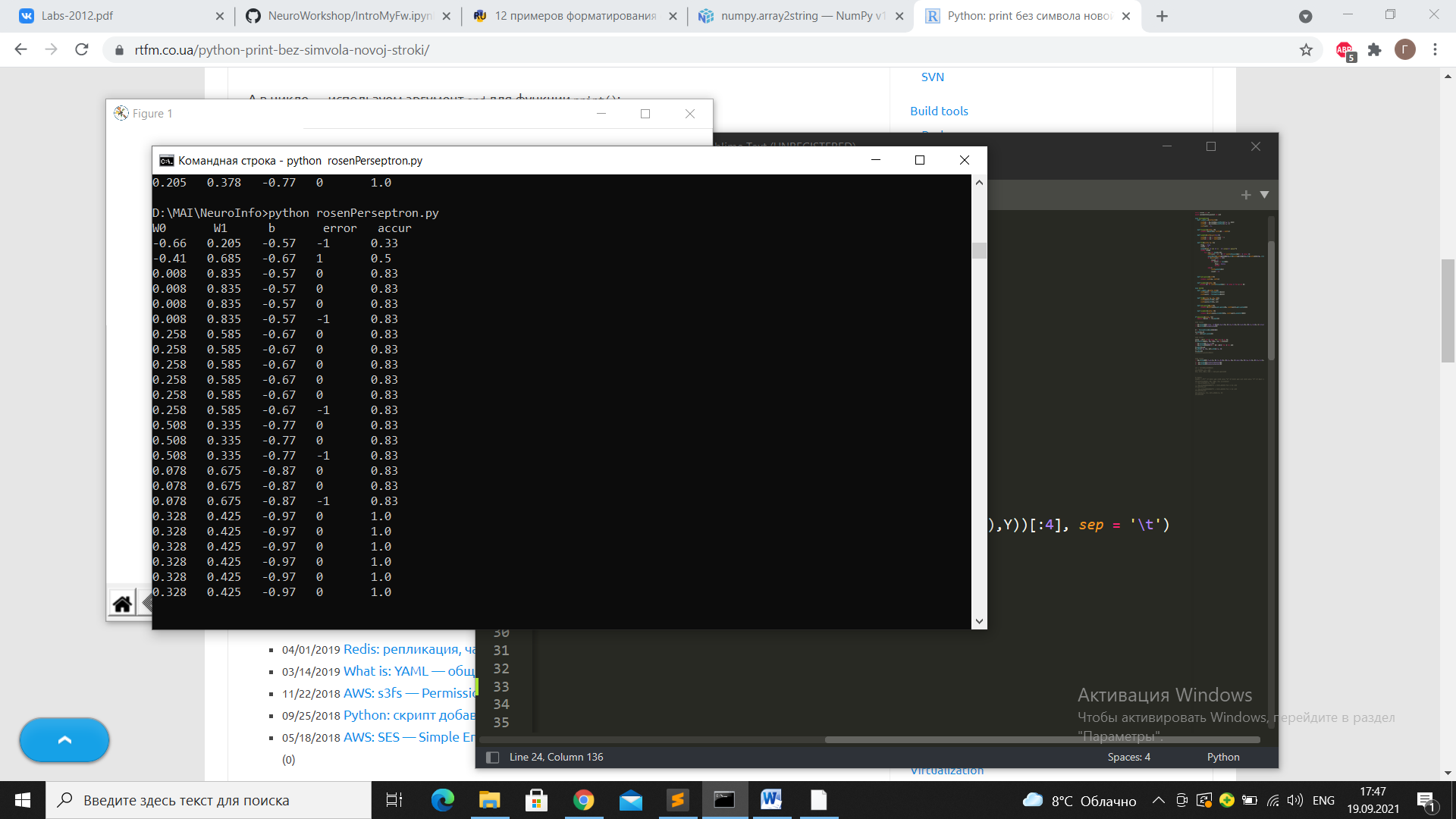
**return** self**.**W**,** self**.**b

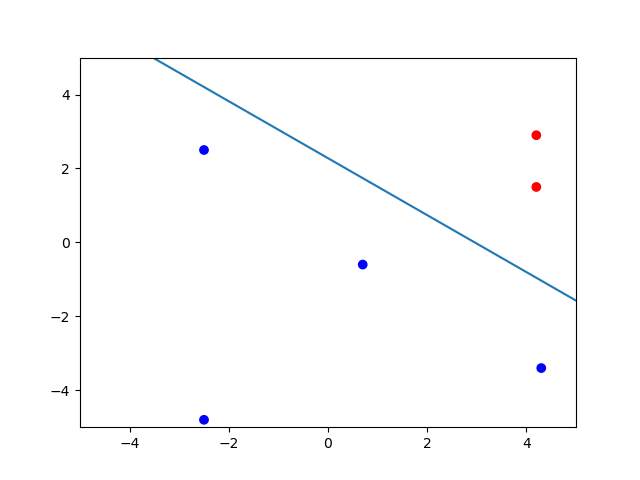
**def** predict**(**self**,** X**):**

**return** **[**0 **if** self**.**forward**(**x**)** **<** 0 **else** 1 **for** x **in** X**]**

сеть: net **=** Perseptron**(len(**X**[**0**]))**

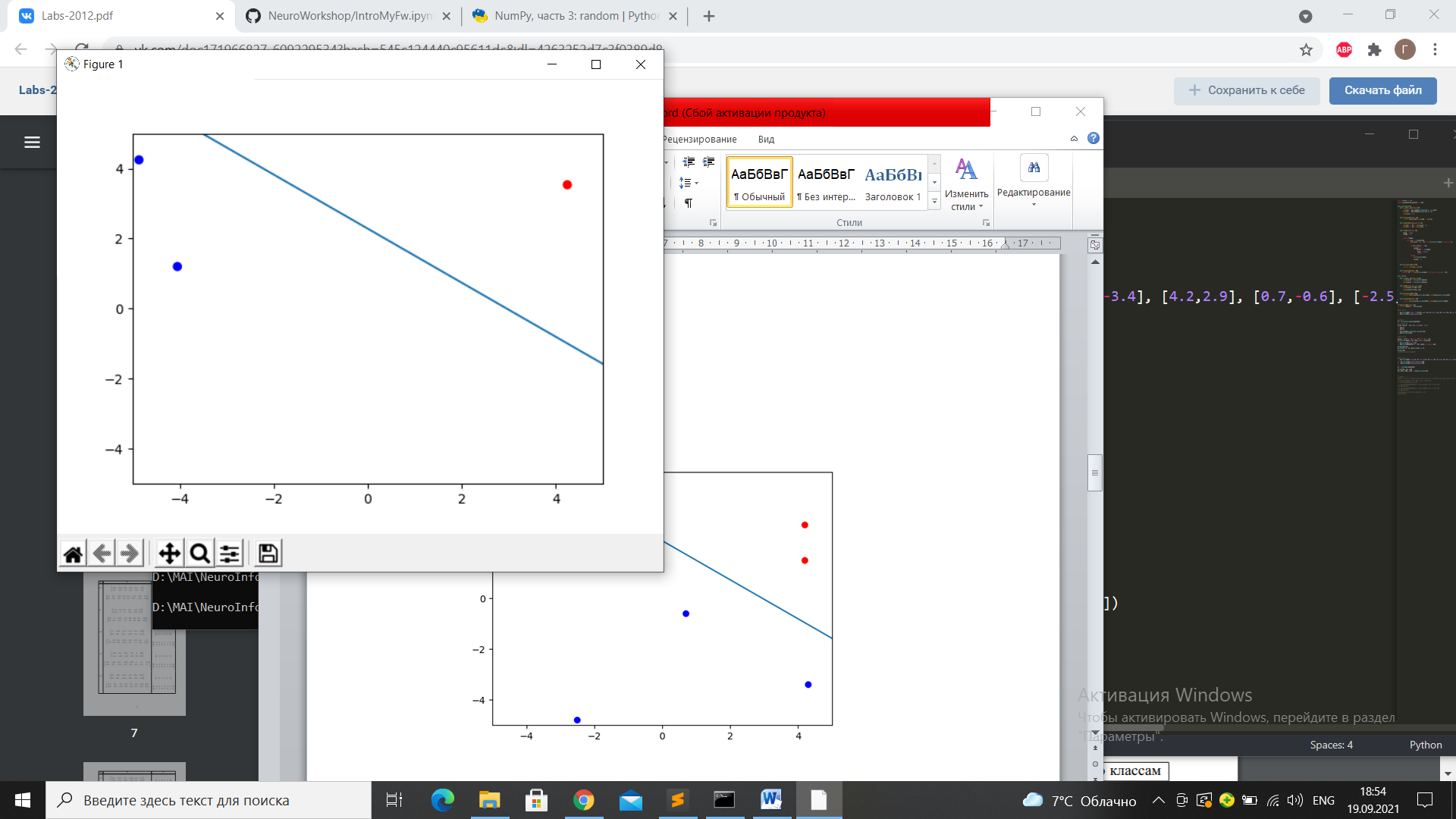
3.3 Процесс обучения





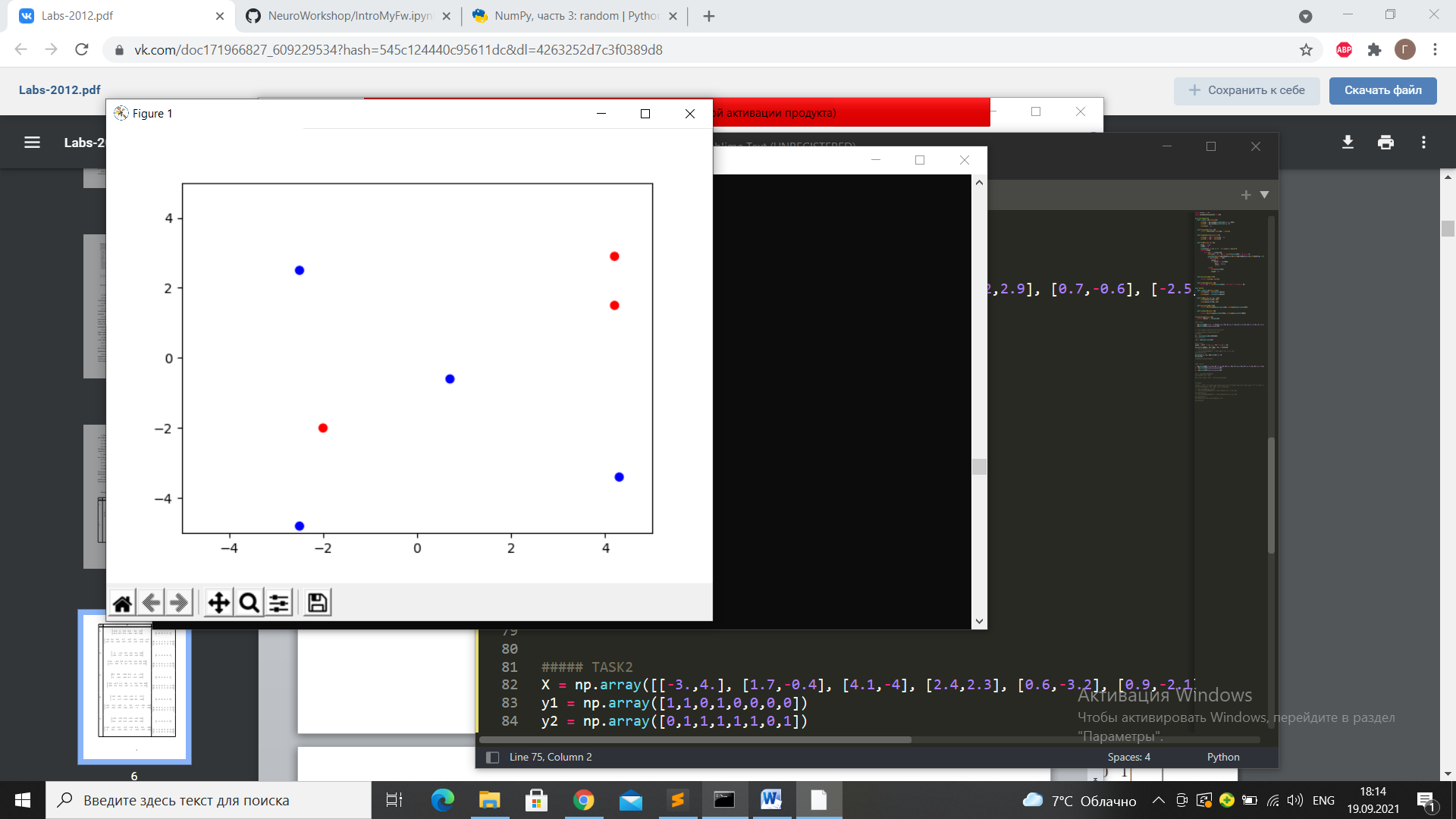
3.4 Проверка качества

Случайным образом выбираем 3 точки с координатами [-5;5] и классифицируем их.



3.5 Неразделимые множества

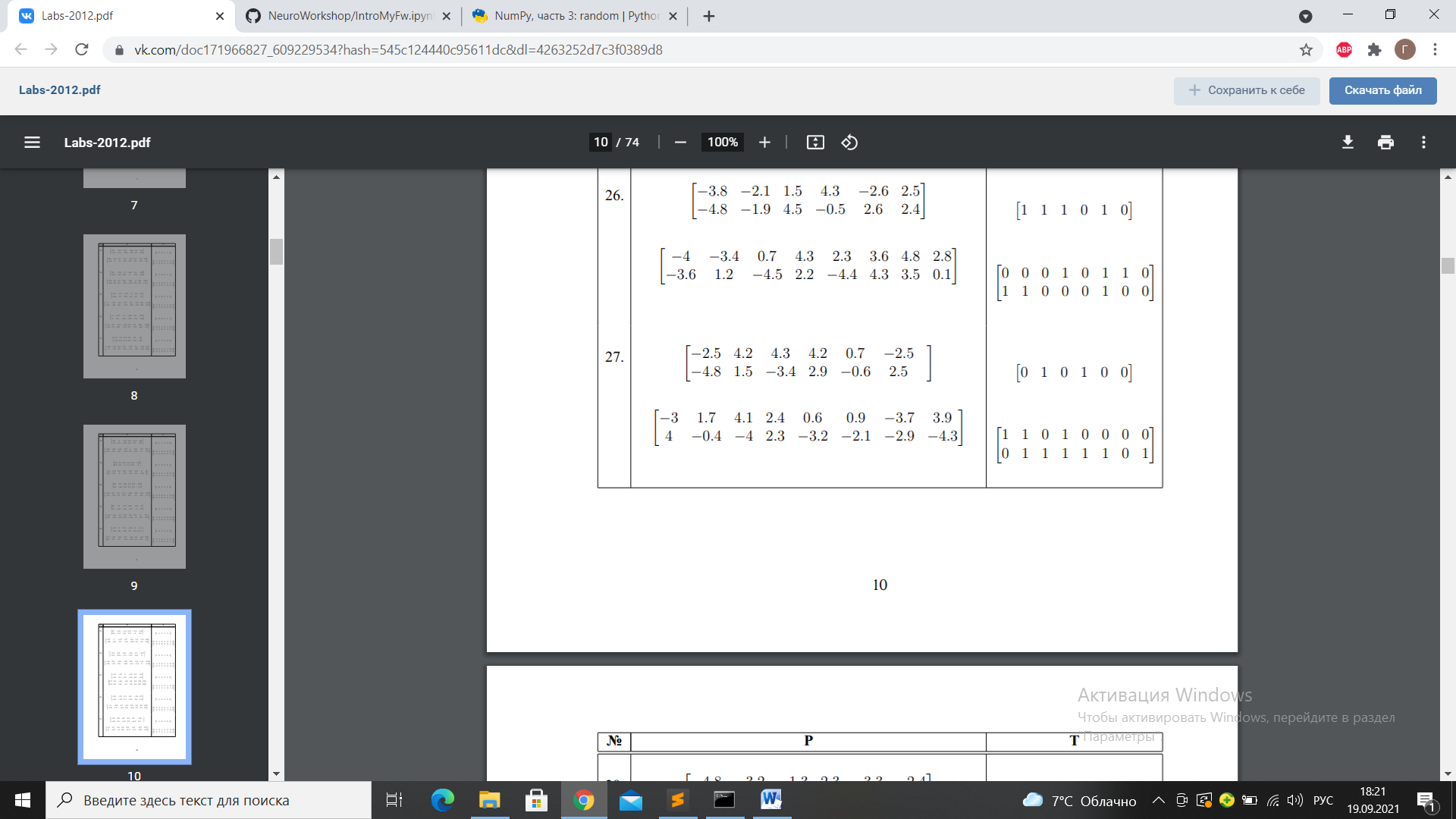
Добавим одну точку так, чтобы множества стали неразделимыми.



Теперь задача не может быть разрешена со 100 процентной точностью, происходит бесконечный цикл.

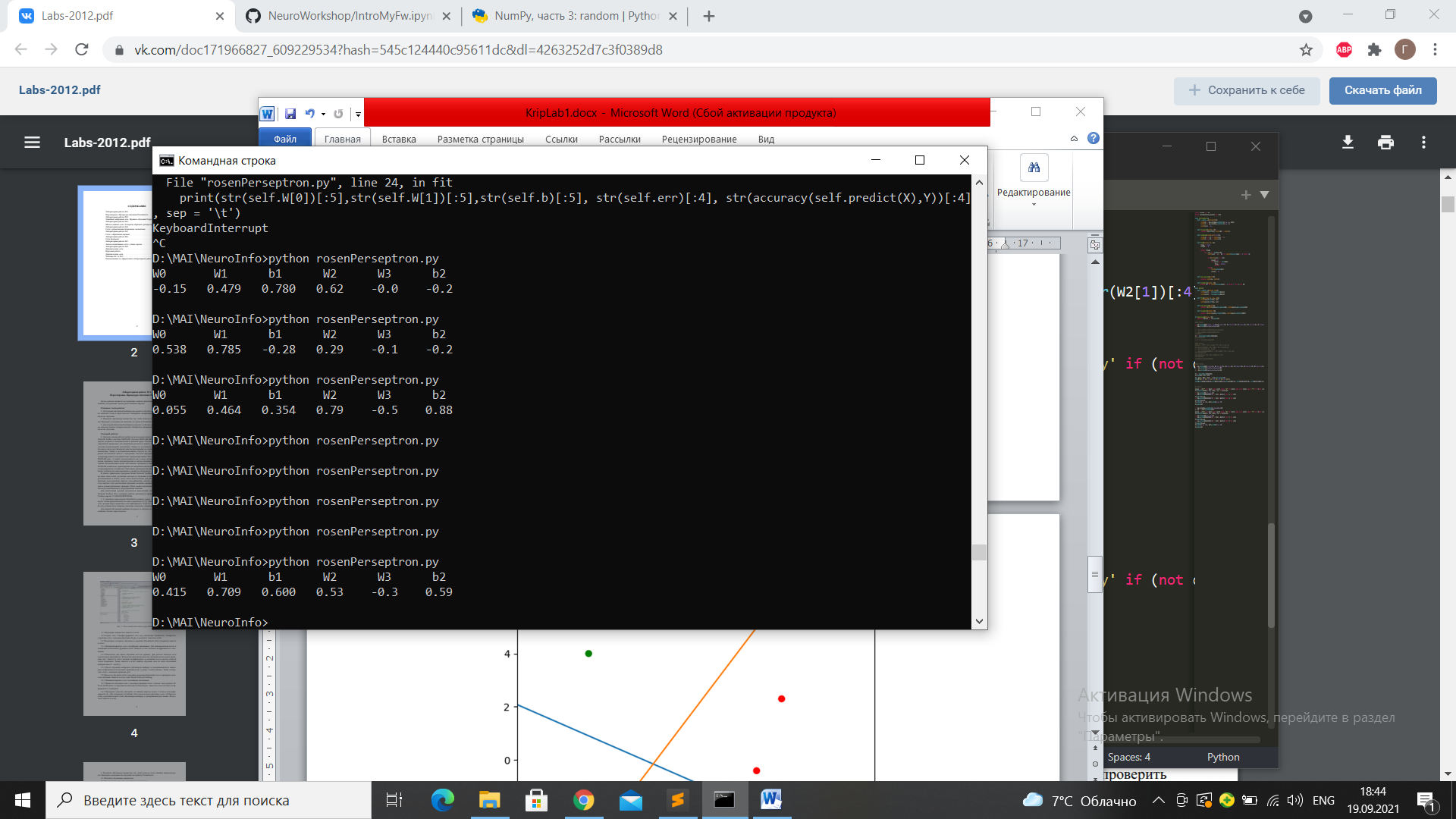
3.6 Классификация по 4 классам

Обучающая выборка

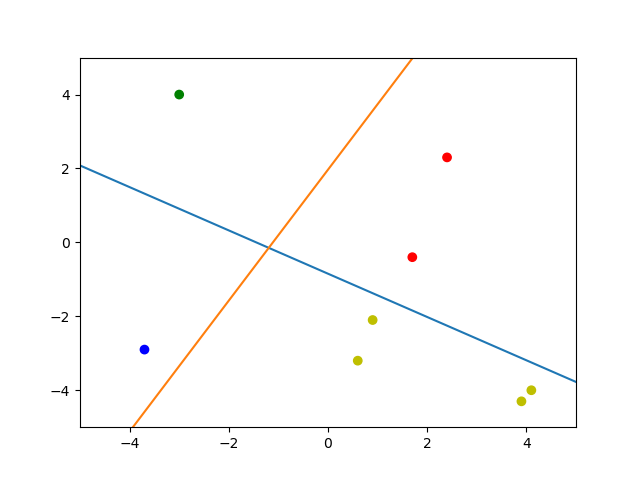


Сеть задается из двух сетей первого задания

Веса после обучения:

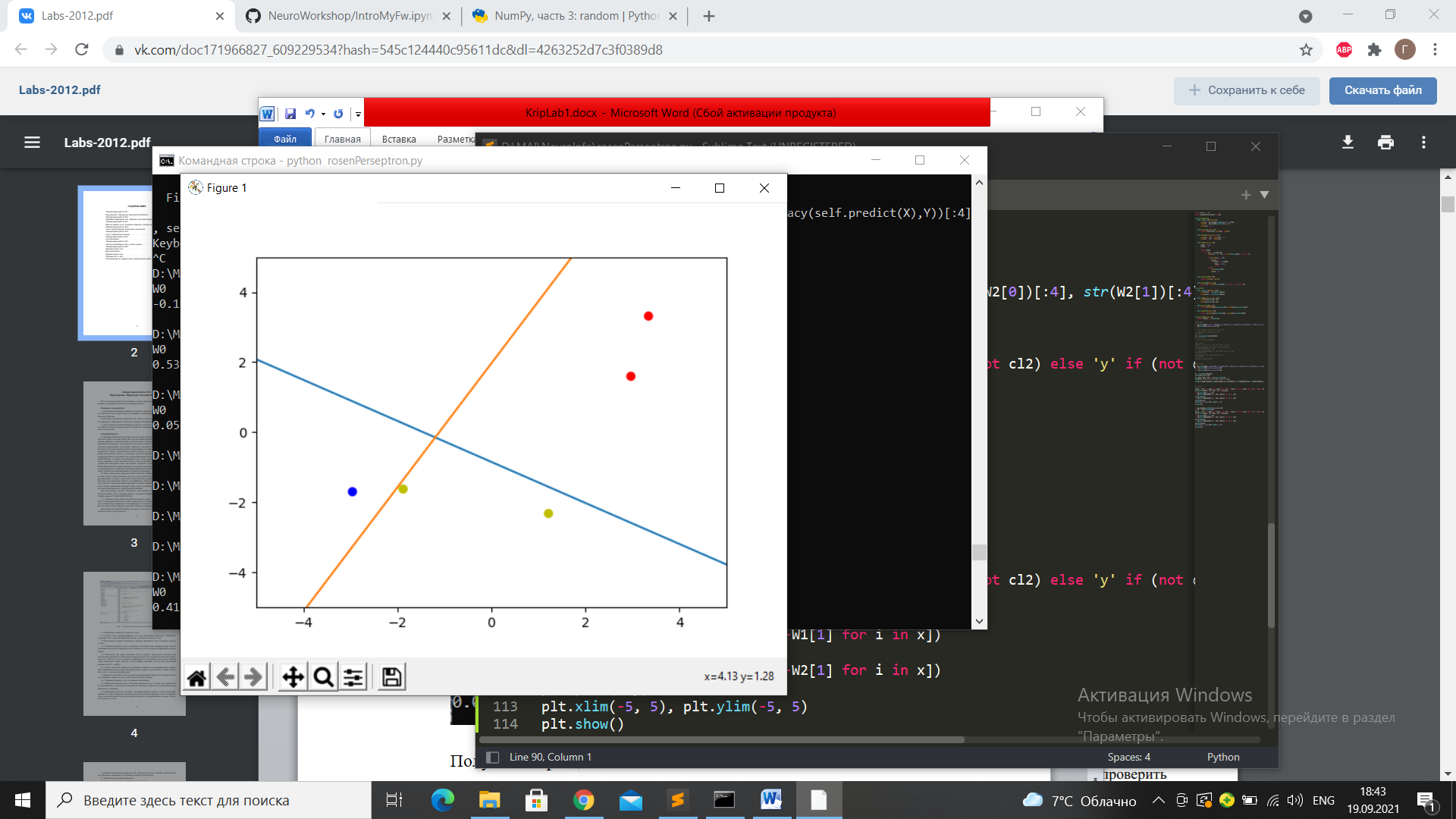


Полученное разделение:



Проверка качества обучения:

Создаем 5 точек с координатами [-5;5] и классифицируем их



1. Выводы

В ходе лабораторной мною была реализована элементарная модель, решающая задачу классификации – персептрон Розенблатта. В качестве функции активации берется пороговая функция, тем самым ошибка может принимать значения только из множества [-1, 0 , 1]. Такая простейшая модель способна решать задачи бинарной классификации с линейно разделимыми множествами.

СПИСОК ЛИТЕРАТУРЫ

1. Beale M., Hagan M., Demuth H. Neural Network Toolbox User’s guide R2011b. The MathWorks, 2011. –pp. 9-3–9-17.

2. Медведев В. С., Потемкин В. Г. Нейронные сети. MATLAB 6/Под общ. ред. к. т. н. В. Г. Потемкина – М.: ДИАЛОГ-МИФИ, 2006. – с. 102–114.

3. Hagan M., Demuth H. Neural Network Design. 1996. – Chapter 4. – 35 pp.