Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

Кафедра информатики

Отчет по лабораторной работе:

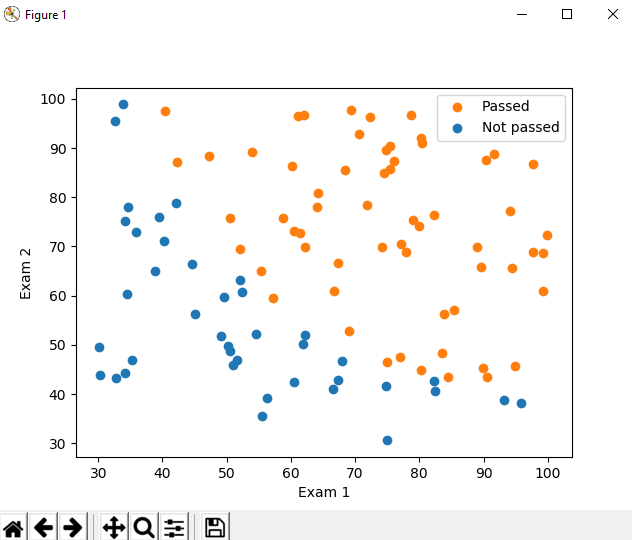
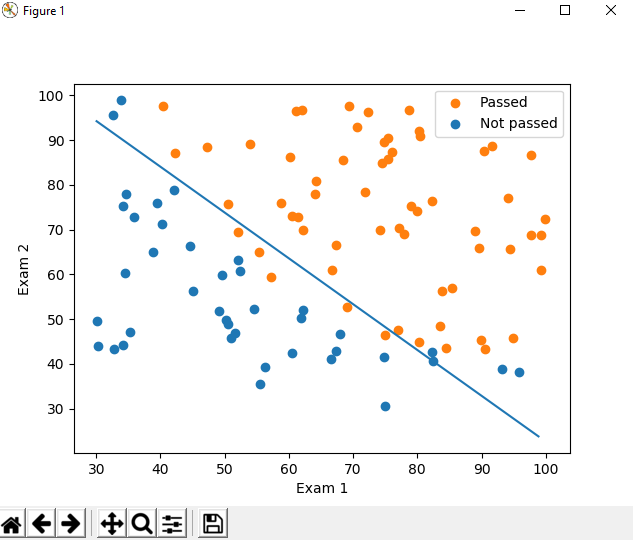
**Лабораторная работа №2 “Логистическая регрессия”**

Выполнил: Карп Александр Игоревич

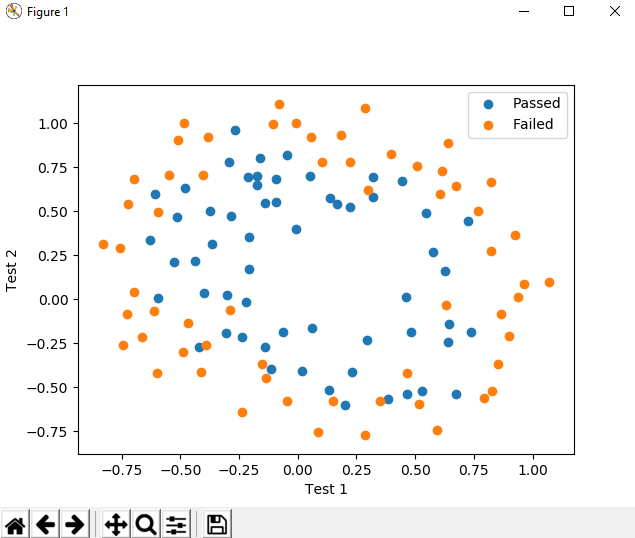
магистрант кафедры информатики

группа №858641

Минск 2019

1. Загрузите данные **ex2data1.txt** из текстового файла.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L61>
2. Постройте график, где по осям откладываются оценки по предметам, а точки обозначаются двумя разными маркерами в зависимости от того, поступил ли данный студент в университет или нет.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L66>  
   
3. Реализуйте функции потерь J(θ) и градиентного спуска для логистической регрессии с использованием векторизации.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L76>
4. Реализуйте другие методы (как минимум 2) оптимизации для реализованной функции стоимости (например, Метод Нелдера — Мида, Алгоритм Бройдена — Флетчера — Гольдфарба — Шанно,   
   генетические методы и т.п.). Разрешается использовать библиотечные реализации методов оптимизации (например, из библиотеки scipy).  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L83>  
     
   Метод Нелдера — Мида:
5. Реализуйте функцию предсказания вероятности поступления студента в зависимости от значений оценок по экзаменам.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L96>
6. Постройте разделяющую прямую, полученную в результате обучения модели. Совместите прямую с графиком из пункта 2.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L99>  
     
   

7) Загрузите данные **ex2data2.txt** из текстового файла.  
<https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L106>  
  
  
8) Постройте график, где по осям откладываются результаты тестов, а точки обозначаются двумя разными маркерами в зависимости от того, прошло ли изделие контроль или нет  
<https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L113>

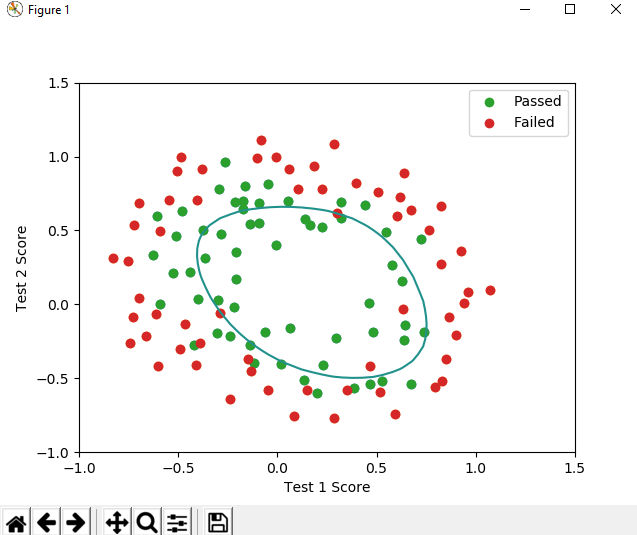


9) Постройте все возможные комбинации признаков x1 (результат первого теста) и x2 (результат второго теста), в которых степень полинома не превышает 6, т.е. 1, x1, x2, x12, x1x2, x22, …, x1x25, x26 (всего 28 комбинаций).  
<https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L125>

10) Реализуйте L2-регуляризацию для логистической регрессии и обучите ее на расширенном наборе признаков методом градиентного спуска.  
<https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L141>

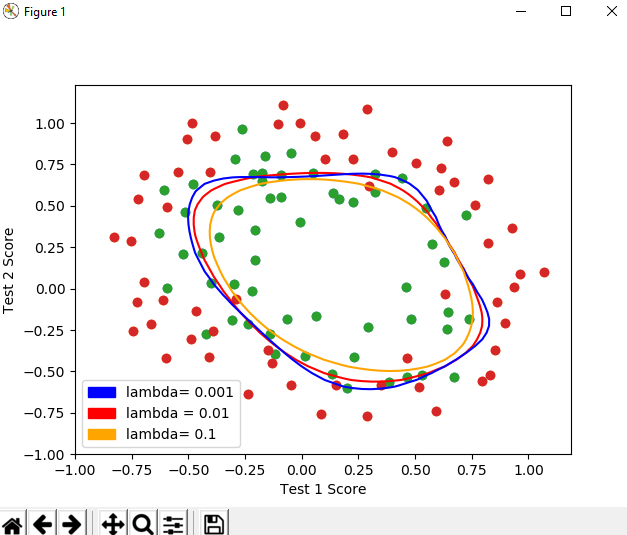
11) Реализуйте другие методы оптимизации.  
<https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L159>

1. Реализуйте функцию предсказания вероятности прохождения контроля изделием в зависимости от результатов тестов.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L174>
2. Постройте разделяющую кривую, полученную в результате обучения модели. Совместите прямую с графиком из пункта 7.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L177>

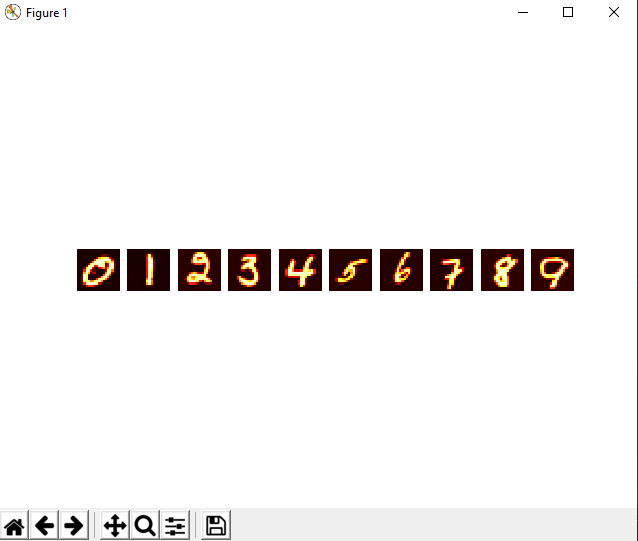


1. Попробуйте различные значения параметра регуляризации λ. Как выбор данного значения влияет на вид разделяющей кривой? Ответ дайте в виде графиков.

<https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L177>



1. Загрузите данные **ex2data3.mat** из файла.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L229>
2. Визуализируйте несколько случайных изображений из набора данных. Визуализация должна содержать каждую цифру как минимум один раз.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L234>



1. Реализуйте бинарный классификатор с помощью логистической регрессии с использованием векторизации (функции потерь и градиентного спуска).  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L249>
2. Добавьте L2-регуляризацию к модели.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L260>
3. Реализуйте многоклассовую классификацию по методу “один против всех”.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L264>
4. Реализуйте функцию предсказания класса по изображению с использованием обученных классификаторов.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L278>
5. Процент правильных классификаций на обучающей выборке должен составлять около 95%.  
   <https://github.com/Karpengold/machine_learning/blob/master/lab2/index.py#L282>