Course › Unit 1 Linear Classifiers and Generalizations (2 weeks) › Project 1: Automatic Review Analyzer ›
1. Introduction

1. Introduction

The goal of this project is to design a classifier to use for sentiment analysis of product reviews. Our training set consists of reviews written by Amazon customers for various food products. The reviews, originally given on a 5 point scale, have been adjusted to a +1 or -1 scale, representing a positive or negative review, respectively.

Below are two example entries from our dataset. Each entry consists of the review and its label. The two reviews were written by different customers describing their experience with a sugar-free candy.

Review	**label**
Nasty No flavor. The candy is just red, No flavor. Just plan and chewy. I would never buy them again	-1
YUMMY! You would never guess that they're sugar-free and it's so great that you can eat them pretty much guilt free! i was so impressed that i've ordered some for myself (w dark chocolate) to take to the office. These are just EXCELLENT!	1

In order to automatically analyze reviews, you will need to complete the following tasks:

1. Implement and compare three types of linear classifiers: the **perceptron** algorithm, the **average perceptron** algorithm, and the **Pegasos** algorithm.

2. Use your classifiers on the food review dataset, using some simple text features.

3. Experiment with additional features and explore their impact on classifier performance.

Setup Details:

For this project and throughout the course we will be using Python 3.6 with some additional libraries. We strongly recommend that you take note of how the NumPy numerical library is used in the code provided, and read through the on-line NumPy tutorial. **NumPy arrays are much more efficient than Python's native arrays when doing numerical computation. In addition, using NumPy will substantially reduce the lines of code you will need to write.**

1. *Note on software: For this project, you will need the NumPy numerical toolbox, and the matplotlib plotting toolbox.*

2. Download sentiment_analysis.tar.gz and untar it into a working directory. The `sentiment_analysis` folder contains the various data files in **.tsv** format, along with the following python files:

   - **project1.py** contains various useful functions and function templates that you will use to implement your learning algorithms.

   - **main.py** is a script skeleton where these functions are called and you can run your experiments.

   - **utils.py** contains utility functions that the staff has implemented for you.

   - **test.py** is a script which runs tests on a few of the methods you will implement. Note that these tests are provided to help you debug your implementation and are not necessarily representative of the tests used for online grading. Feel free to add more test cases locally to further validate the correctness of your code before submitting to the online graders in the codeboxes.

Tip: Throughout the whole online grading system, you can assume the NumPy python library is already imported as `np`. In some problems you will also have access to python's `random` library, and other functions you've already implemented.

This project will unfold both on MITx and on your local machine. You are welcome to implement functions locally and run **test.py** to validate basic functionality, and then copy+paste your code into the MITx code boxes to fully check correctness and receive your grade for individual function implementations. Alternatively, you can also implement the functions online first and after finishing, copy+paste the solution to your local **project1.py** file. Be wary of the number of attempts you have for each problem, especially if you choose the second development flow.

How to Test Locally: In your terminal, navigate to the directory where your project files reside. Execute the command `python test.py` to run all the available tests.

How to Run your Project 1 Functions Locally: In your terminal, enter `python main.py`. You will need to uncomment/comment the relevant code as you progress through the project.

Discussion

Hide Discussion

Topic: Unit 1 Linear Classifiers and Generalizations (2 weeks):Project 1: Automatic Review Analyzer / 1. Introduction

Add a Post

Show all posts by recent activity

? solved - ModuleNotFoundError for matplotbib 1

? File Open 3
Which app should be used to open this file? I tried pycharm it did not work and other then th...

💬 [STAFF] Help Coronavirus outbreak in Italy
Hi, I live in Italy in the region (Lombardia) hit by corona virus outbreak. I got in contact with th...
12

💬 test.py => gets lots of FAIL messages.
Am I supposed to see all PASS, instead of lots of FAIL??
3

💬 Requesting for a deadline extension
Hello Staff, I would just like to bring your kind attention that I am currently pursuing my BE a...
2

? [STAFF ATTENTION] Conflict
Staff, Similar to other students in this class, I am taking the Probability course (MIT 6.431) whi...
5

💬 ANSI color in Windows 10 cmd
If you get weird **[1;32m** and **[m** before and after PASS, FAIL, WARN head while runni...
2

Learn About Verified Certificates