# Homework 4: Logistic Regression

| Contact TA | Yunda Liu |
| --- | --- |
| Released date | Thursday, Oct. 5th, 2023 |
| Submission deadline | Midnight 11:59 PM, Thursday Oct. 12nd, 2023 |
| Total Marks | 100 |

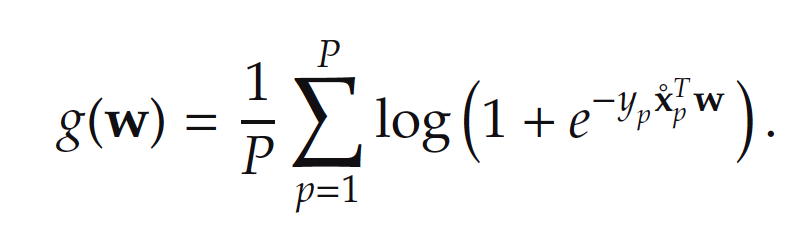
## Task Descriptions

Complete the tasks by filling in the TODOs of the provided code template: [hw4.py](https://drive.google.com/file/d/14myJ2leXimOUYikB91EP_fcj8TSSUOEd/view?usp=sharing).

Feel free to define additional functions such as the model function, the cost function, and python functions that help you plot if they make your code cleaner. You should also reuse your gradient descent function from previous homework.

### Task 1: Implementing the linear classification with Softmax cost

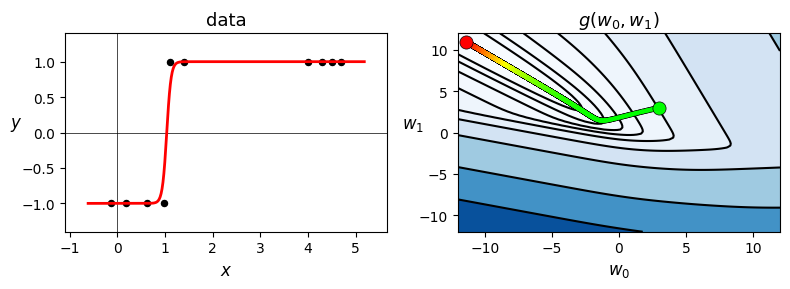
Code up the linear classification with the Softmax cost function shown in Equation below:



Your linear classification will use this 2d classification data 2d\_classification\_data\_v1.csv

(downloaded [here](https://drive.google.com/file/d/17_i2PHWRnMIQduJVpz22i0RaOTq6t0y7/view?usp=sharing)).

Verify that your implementation is working properly by reproducing the final fit shown in the left picture below. To be more specific, the black dots correspond to data points. The red curve corresponds to the fit with the tanh approximation.



You can initialize the weight at w = [3, 3], use max\_iters=2000, and alpha=1.0. No need to normalize the gradient.

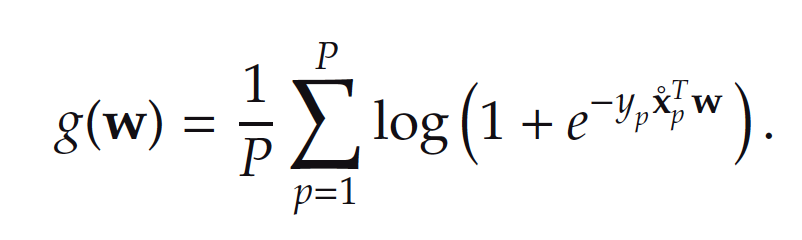
Your final result should produce zero misclassifications.

### Task 2: Compare the efficacy of two-class cost functions

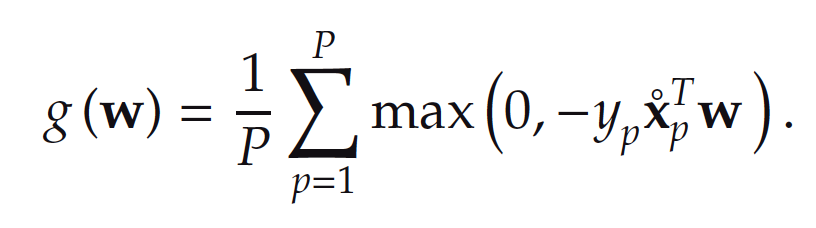
Compare the efficacy of the Softmax and the Perceptron cost functions in terms of the minimal number of misclassifications each can achieve by proper minimization via gradient descent on a breast cancer dataset. Download the dataset [here](https://drive.google.com/file/d/1FbEKMaOh_pNYBz-IdNlTGsWOLSc7Q_ao/view?usp=sharing). The dataset has already been pre-processed so that you can directly learn a linear classification model on it.

This dataset consists of P = 699 data points, each point consisting of N = 9 input of attributes of a single individual and output label indicating whether or not the individual does or does not have breast cancer. You should be able to achieve around 20-40 misclassifications with each method.

Softmax cost:



Perceptron cost:



## If you need some background on the breast cancer dataset, you can find the description of the breast cancer dataset [here](https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original)). But please don’t use the dataset from the official website as the raw dataset is not preprocessed. The input data points are stacked *column-wise* in this dataset, with the final row being the label of each point.

## Deliverables

You submission should contain

* **The completed source code hw4.py**
* **A pdf report**

The source code should be able to run by executing the command: python hw4.py

The PDF report should include:

* Task 1:
  + Report both the accuracy and the misclassification of your model.
  + A figure that shows cost history over iterations. In the figure, the X-axis is the number of iterations, and the Y-axis is the cost
  + A figure that shows the original data and the fitted tanh curve. This figure should look like the one in this requirement. In this figure, you should clearly indicate which is the original data and which is the fitted tanh curve.
* Task 2:
  + Describe the initial values you choose for weight, alpha, and max iterations, and provide the rationale behind your selection.
  + Report both the accuracies and the misclassifications obtained from the two cost functions separately.
  + A figure that compares the cost over iterations between two cost functions. In the figure, you should clearly indicate which line represents each cost function.
  + A figure that compares the accuracy over iterations between two cost functions. In the figure, you should clearly indicate which line represents each cost function.