Tumor Classification using Logistic regression

Sunnam Ravikiran
Department of Computer Science
University at buffalo
buffalo, NY 14221
rsunnam@buffalo.edu

Abstract

Logistic regression is a classification algorithm used to assign observations to a discrete set of classes. The purpose of the project is to build two class logistic regression model to predict cancer cell as Benign (class 0) or Malignant (class 1). The dataset used is the Wisconsin Diagnostic Breast Cancer data set. The data set is split into three portions on which model is trained, validated and tested. It was observed that the model predicted the classes of the cell with good accuracy.

1 Introduction

Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). The goal of logistic regression is to find the best fitting model to describe the relationship between the dichotomous characteristic of interest (dependent variable = response or outcome variable) and a set of independent (predictor or explanatory) variables. Logistic regression generates the coefficients which can be used to find the output variables. Logistic regression transforms its output using the logistic sigmoid function to return a probability value which can then be mapped to two or more discrete classes

2 Dataset

Wisconsin Diagnostic Breast Cancer (WDBC) dataset is used for training, validation and testing. The dataset contains 569 instances with 32 attributes (ID, diagnosis (B/M), 30 real-valued input features). Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. Computed features describes the following characteristics of the cell nuclei present in the image:

1	radius (mean of distances from center to points on the perimeter)
2	texture (standard deviation of gray-scale values)
3	perimeter
4	area
5	smoothness (local variation in radius lengths)
6	compactness (perimeter2/area − 1.0)

7	concavity (severity of concave portions of the contour)
8	concave points (number of concave portions of the contour)
9	symmetry
10	fractal dimension ("coastline approximation" - 1)

The Given dataset is partitioned into training, validation and testing data. Randomly choose 80% of the data for training and the rest for validation and testing.

```
The training dataset contains [455] instances.
The validation dataset contains [57] instances.
The testing dataset contains [57] instances.
```

3 Preprocessing the dataset

The given Wisconsin Diagnostic Breast Cancer (WDBC) dataset has been preprocessed before using logistic regression model as follows.

- *Processing the data file (CSV)*
    1. The CSV data file is read and processed to identify the dependent and independent valuables.
    2. The dependent valuable is identified by labels [M - Malignant, B- Benign] where Malignant is treated as (class 1) and Benign is treated as (class 0). And respectively labels are encoded.
    3. The independent variables which don't contribute the prediction of the dependent variable are dropped from the dataset (patient Id).

- *Data Partitioning*

  In machine learning we usually split our data into three subsets: train, validate and test dataset, and fit our model on the train data, in order to make predictions on the test data. This is done to avoid the overfitting of the model.

  The Given dataset is partitioned into training, validation and testing data. Randomly choose 80% of the data for training and the rest 10% for validation and remaining testing. The dataset after splitting is as follows.

  ```
  1. The training dataset contains [455] instances.
  2. The validation dataset contains [57] instances.
  3. The testing dataset contains [57] instances.
  ```

- *Normalization*

  Feature scaling is a method used to normalize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step. The goal of normalization is to change the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values. All the datasets (train, validation and test) have been mean normalized.

4 Architecture

Logistic regression comes from the fact that linear regression can also be used to perform classification problem but the logistic regression is not linear (because it involves a transformation with both an exponential function of x and a ratio.

$$\sum_j \theta_j x_j = \theta^\top x.$$

The output of liner regression is put through sigmoid function to classify the output into class 1 or class 0.

4.1 Sigmoid activation

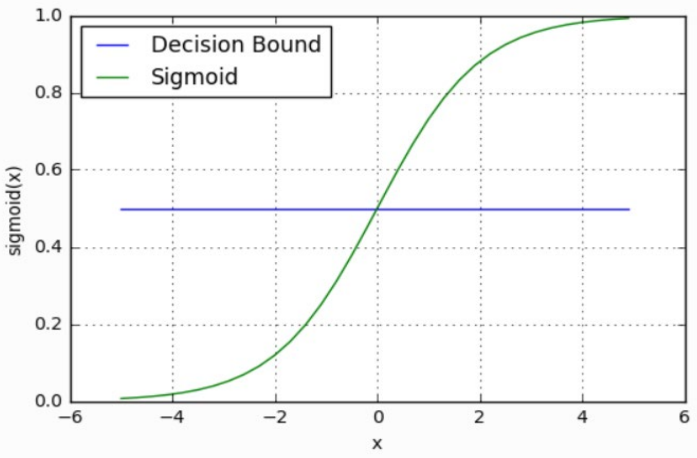
In order to map predicted values to probabilities, we use the sigmoid function. The function maps any real value into another value between 0 and 1. In machine learning, we use sigmoid to map predictions to probabilities.

$$P(y=1|x) = h_\theta(x) = \frac{1}{1+\exp(-\theta^\top x)} \equiv \sigma(\theta^\top x),$$
$$P(y=0|x) = 1 - P(y=1|x) = 1 - h_\theta(x).$$

4.1.1 Decision boundary

Our current prediction function returns a probability score between 0 and 1. In order to map this to a discrete class (class 0/ class1) we select a threshold value or tipping point above which we will classify values into class 1 and below which we classify values into class 2.

$$p \geq 0.5, class = 1$$
$$p < 0.5, class = 0$$



4.2 Cost Function [Cross-entropy]

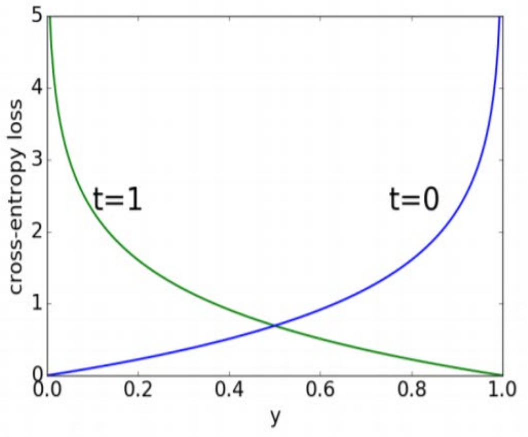
Cross-entropy loss, or log loss, measures the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss increases as the predicted probability diverges from the actual label.

$$J(\theta) = -\sum_{i} \left(y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \right).$$

the cost function penalizes confident and wrong predictions more than it rewards confident and right predictions! The corollary is increasing prediction accuracy (closer to 0 or 1) has diminishing returns on reducing cost due to the logistic nature of our cost function.

$$\begin{cases} -\log y & \text{if } t = 1 \\ -\log 1 - y & \text{if } t = 0 \end{cases}$$

$$-t \log y - (1 - t) \log 1 - y$$



4.3 Gradient Descent

Gradient descent is an optimization algorithm used to minimize some function by iteratively moving in the direction of steepest descent as defined by the negative of the gradient. In machine learning, we use gradient descent to update the parameters of our model. Parameters refer to coefficients in Linear Regression and weights in neural networks.

4.3.1 Learning rate

The size of these steps is called the *learning rate*. With a high learning rate we can cover more ground each step, but we risk overshooting the lowest point since the slope of the hill is constantly changing. With a very low learning rate, we can confidently move in the direction of the negative gradient since we are recalculating it so frequently. A low learning rate is more precise, but calculating the gradient is time-consuming, so it will take us a very long time to get to the bottom.

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \sum_i x_j^{(i)} (h_\theta(x^{(i)}) - y^{(i)}).$$

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} \qquad \text{for j := 0...n}$$

4.4 Results

The preprocessing of data along with logistic regression implementation is done in python. the dataset is split into train, validation and test.

The initial [θ1, θ2, θ3….] have been initialized to zero and the values have been calculated iteratively using gradient decent approach. For each epoch the training loss and validation is calculated and plotted.

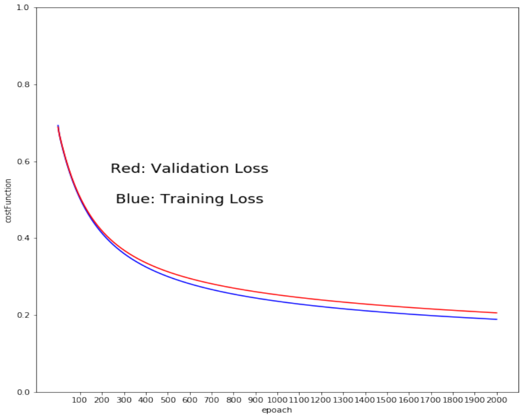
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

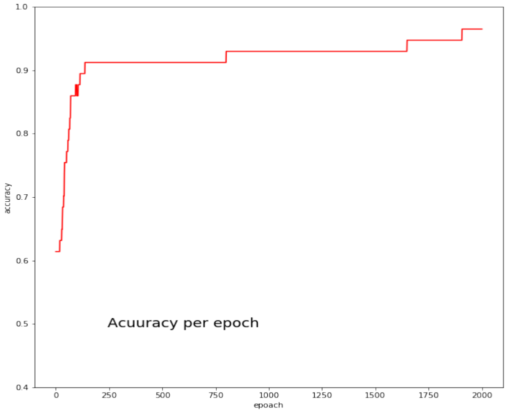
$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

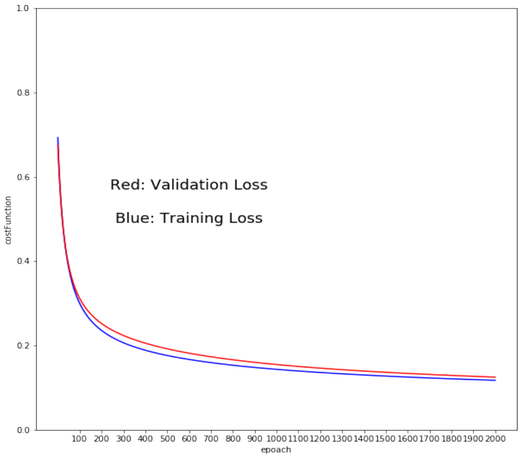
Learning rate		
0.1	Accuracy	0.96491228
	Recall	0.91666667
	Precision	1.0
0.5	Accuracy	0.98245614
	Recall	0.95652174
	Precision	1.0

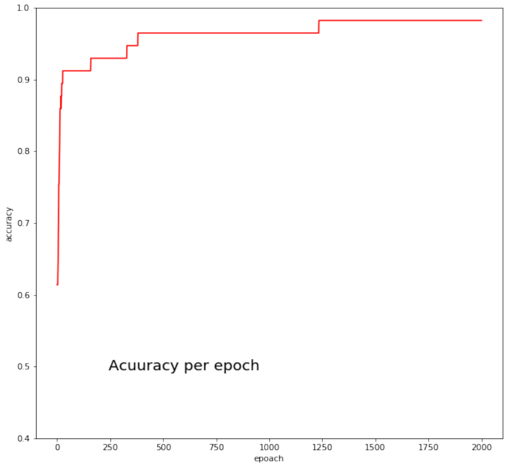
For alpha 0.1





For alpha 0.5





5 Conclusion

Logistic regression is a well know approach for a two class classification. The accuracy, recall and precision can be seen from the above table. We can also observe that by tuning the hyper parameters we can achieve better results. In the above project the number of epoch are put constant and the learning rate is increased, this lead to a better accuracy as the cost function is minimized even further. The model with hyper parameter 0.5 was able to achieve a accuracy of 98 percent.