S21CSE471: Statistical Methods in AI **Spring 2021**

Logistic Regression

Prepared by:

Abhisek Mohapatra (2020201020), Aditya Rathi(2020201041), Anchal Soni(2020201099)

1 What is Logistic regression

In statistics, the logistic model (or logit model) is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. This can be extended to model several classes of events such as determining whether an image contains a cat, dog, lion, etc. Each object being detected in the image would be assigned a probability between 0 and 1, with a sum of one.

2 What is the Classification Problem?

Consider a scenario where we need to classify whether an email is spam or not. If we use linear regression for this problem, there is a need for setting up a threshold based on which classification can be done. Say if the actual class is malignant, predicted continuous value 0.4 and the threshold value is 0.5, the data point will be classified as not malignant which can lead to serious consequence in real time.

From this example, it can be inferred that linear regression is not suitable for classification problem. Linear regression is unbounded, and this brings logistic regression into picture. Their value strictly ranges from 0 to 1.

Logistic regression is the go-to method for binary classification problems (problems with two class values).

3 Why is it logistic regression?

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).

Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled ”0” and ”1”. In the logistic model, the log-odds (the logarithm of the odds) for the value labeled ”1” is a linear combination of one or more independent variables (”predictors”); the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value).

The corresponding probability of the value labeled ”1” can vary between 0 (certainly the value ”0”) and 1 (certainly the value ”1”), hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name. The unit of measurement for the log-odds scale is called a logit, from **log**istic un**it**, hence the alternative names.

4 Sigmoid Function

The Sigmoid function is a mathematical function which has a characteristic S-shaped curve. All sigmoid functions have the property that they map the entire input set into a small range such as between 0 and 1, or -1 and 1 such that the sigmoid function can be used to convert a real value into one that can be interpreted as a probability.

All sigmoid functions are monotonic and have a bell-shaped first derivative. There are several sigmoid functions and some of the best-known are the logistic function, the hyperbolic tangent, and the arctangent. All share the same basic S shape.

One of the most widely used sigmoid functions is the **logistic function**, which maps any real value to the range (0, 1).

The logistic function can be graphically represented as:

$$\sigma(\mathbf{x}) = \frac{\mathbf{1}}{\mathbf{1 + e^{-x}}}$$

4.1 Graphical Representation

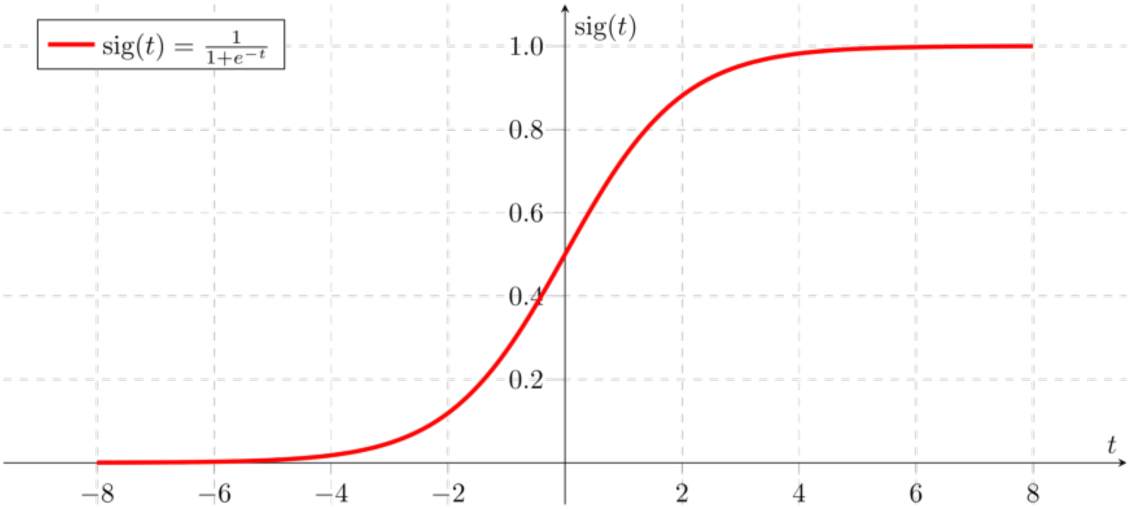


Figure 1: Graphical Representation of Logistic Regression

As the value of n gets larger, the value of the sigmoid function gets closer and closer to 1 and as n gets smaller, the value of the sigmoid function is get closer and closer to 0

5 Derivative of Sigmoid Function

We have,

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

We know, derivative of Logistic function $= \sigma'(x) = \frac{d}{dx}\sigma(x)$

$$= \frac{d}{dx}\frac{1}{1+e^{-x}} = \frac{d}{dx}\left(1+e^{-x}\right)^{-1}$$

$$= \frac{d}{dx}\left(1+e^{-x}\right)^{-1} = -\left(1+e^{-x}\right)^{-2} \cdot \frac{d}{dx}\left(1+e^{-x}\right)$$

Next, according to Rule of Linearity which says that:

$$[a \cdot u(x) + b \cdot v(x)]' = a \cdot u'(x) + b \cdot v'(x)$$

we get,

$$\implies -\left(1+e^{-x}\right)^{-2} \cdot \frac{d}{dx}\left(1+e^{-x}\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(\frac{d}{dx}[1] + \frac{d}{dx}\left[e^{-x}\right]\right)$$

$$\implies -\left(1+e^{-x}\right)^{-2} \cdot \left(\frac{d}{dx}[1] + \frac{d}{dx}\left[e^{-x}\right]\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(0 + \frac{d}{dx}\left[e^{-x}\right]\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(\frac{d}{dx}\left[e^{-x}\right]\right)$$

Now, Applying exponential rule, we get:

$$\implies -\left(1+e^{-x}\right)^{-2} \cdot \left(\frac{d}{dx}\left[e^{-x}\right]\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(e^{-x} \cdot \frac{d}{dx}[-x]\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(e^{-x} \cdot -\frac{d}{dx}[x]\right)$$

$$= -\left(1+e^{-x}\right)^{-2} \cdot \left(\frac{d}{dx}\left[e^{-x}\right]\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(e^{-x} \cdot \frac{d}{dx}[-x]\right) = -\left(1+e^{-x}\right)^{-2} \cdot \left(e^{-x} \cdot -1\right)$$

since,

$$\frac{d}{dx}[x] = 1$$

$$\implies -\left(1+e^{-x}\right)^{-2} \cdot \left(e^{-x} \cdot -1\right) = \left(1+e^{-x}\right)^{-2} \cdot e^{-x}$$

which can be expressed as:

$$\left(1+e^{-x}\right)^{-2} \cdot e^{-x} = \frac{e^{-x}}{\left(1+e^{-x}\right)^{2}}$$

We can further rewrite the above equation in the context of application in Machine Learning as follows:

$$\implies \frac{e^{-x}}{\left(1+e^{-x}\right)^{2}} = \frac{1 \cdot e^{-x}}{\left(1+e^{-x}\right) \cdot \left(1+e^{-x}\right)}$$

And then rewrite as:

$$\implies \frac{1 \cdot e^{-x}}{\left(1+e^{-x}\right) \cdot \left(1+e^{-x}\right)} = \frac{1}{\left(1+e^{-x}\right)} \cdot \frac{e^{-x}}{\left(1+e^{-x}\right)}$$

$$\implies \frac{1}{\left(1+e^{-x}\right)} \cdot \frac{e^{-x}}{\left(1+e^{-x}\right)} = \frac{1}{\left(1+e^{-x}\right)} \cdot \frac{e^{-x}+1-1}{\left(1+e^{-x}\right)}$$

And now let's break the fraction and rewrite it as:

$$\implies \frac{1}{\left(1+e^{-x}\right)} \cdot \frac{e^{-x}+1-1}{\left(1+e^{-x}\right)} = \frac{1}{\left(1+e^{-x}\right)} \cdot \left(\frac{1+e^{-x}}{1+e^{-x}} - \frac{1}{1+e^{-x}}\right)$$

Cancelling out the numerator and denominator

$$\implies \frac{1}{\left(1+e^{-x}\right)} \cdot \left(\frac{1+e^{-x}}{1+e^{-x}} - \frac{1}{1+e^{-x}}\right) = \frac{1}{\left(1+e^{-x}\right)} \cdot \left(1 - \frac{1}{1+e^{-x}}\right)$$

Finally, we can express the derivative of logistic function as:

$$\implies \frac{1}{(1+e^{-x})}\cdot\left(1-\frac{1}{1+e^{-x}}\right)=\sigma(\mathbf{x})\cdot(\mathbf{1}-\sigma(\mathbf{x}))$$

that is,

$$\sigma'(x)=\sigma(\mathbf{x})\cdot(\mathbf{1}-\sigma(\mathbf{x})) \tag{1}$$

6 Prediction By Model

Let us assume that the output predicted by our ML model is y_{pred}

Now, this predicted value when passed through sigmoid function produces following output:

$$\sigma(y_{pred})=\frac{1}{1+e^{-y_{pred}}}=y_{final}$$

We know,

$$y_{pred}=w^T.x$$

where,

x is the input vector and w is the weight matrix.

Therefore,

$$\sigma(w^T.x)=\frac{1}{1+e^{-w^T.x}}$$

Now, if we use Mean Squared Error Cost function, we get :

$$J(w)=\frac{1}{2m}\sum_{i=1}^{m}\left(y_{(i)}-\hat{y}_{(i)}\right)^2$$

$$\implies J(w)=\frac{1}{2m}\sum_{i=1}^{m}\left(y_{(i)}-\frac{1}{1+e^{-y_{pred}}}\right)^2$$

$$\implies J(w)=\frac{1}{2m}\sum_{i=1}^{m}\left(y_{(i)}-\frac{1}{1+e^{-w^T.x}}\right)^2$$

$$\implies J(w)=\frac{1}{2m}\sum_{i=1}^{m}\left(y_{(i)}-\sigma(w^T.x)\right)^2$$

Now, the above function $J(w)$ is not a convex function with many local minimums since our prediction function is a non-linear function due to sigmoid function.

If our cost function has many local minimums, then gradient descent may not find the optimal global minimum. So to choose values for the parameters of logistic regression, we use maximum likelihood estimation (MLE).

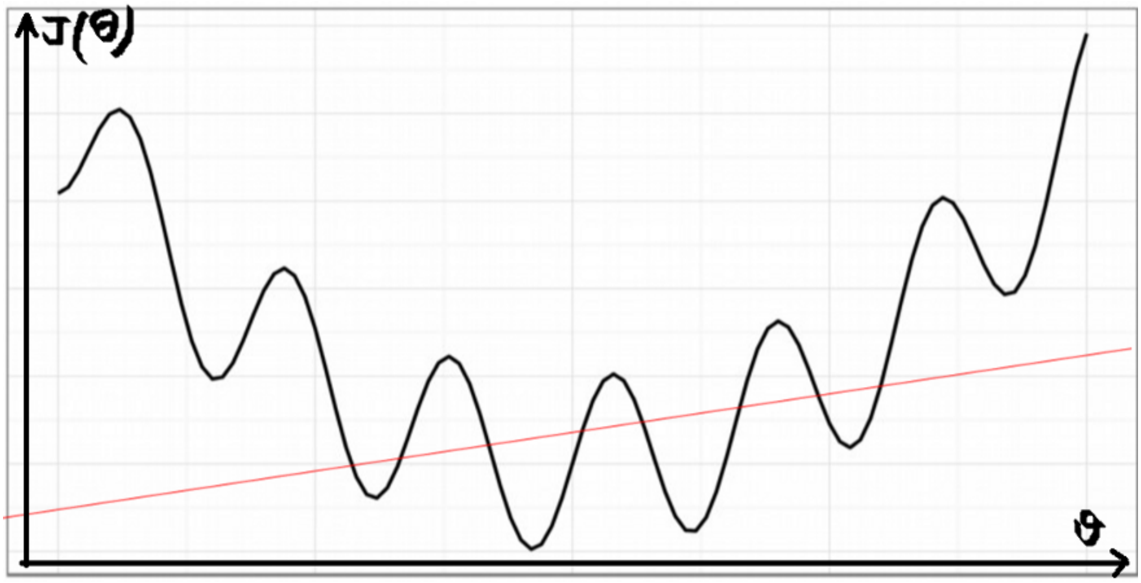


Figure 2: Graphical Representation of Non-Convex function

7 Maximum Likelihood Estimate

Maximum likelihood estimation is a method that determines values for the parameters of a model. The parameter values are found such that they maximise the likelihood that the process described by the model, produced the data that were actually observed. The values that we find are called the maximum likelihood estimates (MLE).

Our aim is to get as close to the actual output, ie we want to maximize the given probabilities

$$P(Y_i = 1||X_i, w) = \sigma(w^T X_i) \tag{2}$$

using Total Probability Theorem

$$P(Y_i = 0||X_i, w) = 1 - \sigma(w^T X_i) \tag{3}$$

We can combine the above 2 equations into a single equation and maximize it, as follows

$$P(Y_i|X_i, w) = (\sigma(w^T X_i))^{Y_i} * (1 - \sigma(w^T X_i))^{(1-Y_i)} \tag{4}$$

Why above equation works?

$$\text{Let } P(Y_i|X_i, w) = term1 * term2$$

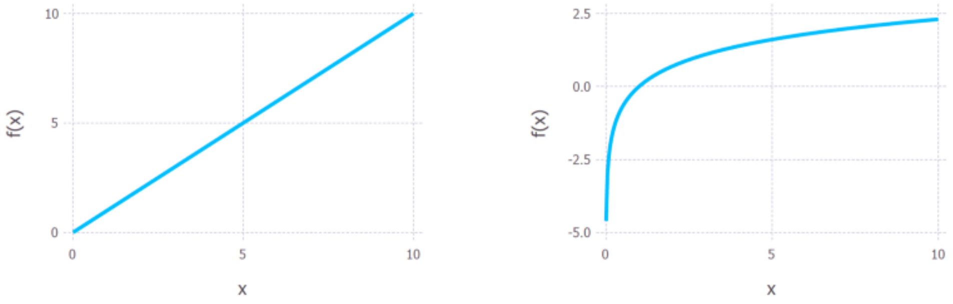
When $Y_i = 0$, term1 goes to zero, so we want to maximize the term2, which means minimizing the term $\sigma(w^T X_i)$ i.e. minimizing the y-axis in the sigmoid function graph and we would be close to predicting 0 (The actual value of Y_i).
Now when $Y_i = 1$, then term2 goes to 0, so we need to maximize the value of $\sigma(w^T X_i)$, which means the predicted value is as close to 1 (The actual value of Y_i)

The above equation - 4 was for one input data point. Our goal is to maximize it for all data points, w.r.t to w,so we have the function

$$L(w) = \prod_{i=1}^{N} P(Y_i|X_i, w) \text{ (Maximizing w.r.t to w)}$$

The log likelihood

The above expression for the total probability is actually quite a pain to differentiate, so it is almost always simplified by taking the natural logarithm of the expression. This is absolutely fine because the natural logarithm is a monotonically increasing function. This means that if the value on the x-axis increases, the value on the y-axis also increases (see Figure-1). This is important because it ensures that the maximum value of the log of the probability occurs at the same point as the original probability function. Therefore we can work with the simpler log-likelihood instead of the original likelihood.



(a) $f(x) = x$

(b) $f(x) = \ln(x)$

Figure 3: (Monotonic behaviour of the original function, y = x on the left and the (natural) logarithm function y = ln(x). These functions are both monotonic because as you go from left to right on the x-axis the y value always increases.)

Taking logs of the original expression gives us:

$$\log\left(L(w)\right) = log(\prod_{i=1}^{N} P(Y_i|X_i, w))$$

Let $G(w) = \log\left(L(w)\right)$

Using the laws of logarithms ie $\log\left(a * b\right) = \log\left(a\right) + \log\left(b\right)$ and $\log\left(a^b\right) = b \times \log\left(a\right)$, we simplify the above expression as:

$$G(w) = \log\left(L(w)\right) = \sum_{i=1}^{N} log(P(Y_i|X_i, w)) \tag{5}$$

Putting value of $P(Y_i|X_i, w)$ from equation - 4, we get

$$G(w) = \sum_{i=1}^{N} log(\sigma(w^T X_i))^{Y_i} * (1 - \sigma(w^T X_i))^{(1-Y_i)}) \tag{6}$$

Using the property $\log\left(a * b\right) = \log\left(a\right) + \log\left(b\right)$, we have:

$$G(w) = \sum_{i=1}^{N} (log(\sigma(w^T X_i))^{Y_i} + log(1 - \sigma(w^T X_i))^{(1-Y_i)}) \tag{7}$$

As $\log\left(a^b\right) = b \times \log\left(a\right)$, we have:

$$G(w) = \sum_{i=1}^{N} ((Y_i) log(\sigma(w^T X_i)) + (1 - Y_i) log(1 - \sigma(w^T X_i))) \tag{8}$$

To find the maxima we differentiate the above equation w.r.t w.

$$G'(w) = \sum_{i=1}^{N}(Y_i.\frac{1}{\sigma(w^T X_i)}\sigma'(w^T X_i).X_i + (1-Y_i).\frac{1}{1-\sigma(w^T X_i)}.(-1).\sigma'(w^T X_i).X_i)$$

$$G'(w) = \sum_{i=1}^{N}(\frac{Y_i}{\sigma(w^T X_i)} - \frac{(1-Y_i)}{1-\sigma(w^T X_i)}).\sigma'(w^T X_i).X_i)$$

As from equation - 1 $\sigma'(w^T x) = \sigma(w^T x).(1-\sigma(w^T x))$, so we have:

$$G'(w) = \sum_{i=1}^{N}(\frac{Y_i}{\sigma(w^T X_i)} - \frac{(1-Y_i)}{1-\sigma(w^T X_i)}).\sigma(w^T X_i)(1-\sigma(w^T X_i)).X_i)$$

$$G'(w) = \sum_{i=1}^{N}(Y_i.(1-\sigma(w^T X_i)).X_i - (1-Y_i).\sigma(w^T X_i).X_i)$$

$$G'(w) = \sum_{i=1}^{N}(Y_i - Y_i.\sigma(w^T X_i) + \sigma(w^T X_i) + Y_i.\sigma(w^T X_i)).X_i$$

$$G'(w) = \sum_{i=1}^{N}(Y_i + \sigma(w^T X_i)).X_i \tag{9}$$

Now, we use **Gradient Ascent** to maximize the function G(w). In gradient ascent, we move in the direction of the gradient. Another option is that we minimize the negative of the function using Gradient Descent. Let λ be the learning rate. So the new value of w is calculated using previous value of w as follows:

$$w_{t+1} = w_t + \lambda(G'(w))$$

8 Classification Rule

Based on the following rules we classify the data point to one class or the other using the threshold limit as 0.5 (Though can be changed if we want to prefer one class more.)

$$\sigma(w^T X) > 0.5 \implies (Class-1)$$
$$\sigma(w^T X) <= 0.5 \implies (Class-2)$$

9 Multiclass classification using logistic regression

Algorithms such as the Perceptron, Logistic Regression, and Support Vector Machines were designed for binary classification and do not natively support classification tasks with more than two classes.

heuristic methods can be used to split a multi-class classification problem into multiple binary classification datasets and train a binary classification model each.

Two examples of these heuristic methods include:

1. One-vs-Rest (OvR)
2. One-vs-One (OvO)

One-Vs-Rest for Multi-Class Classification

It involves splitting the multi-class dataset into multiple binary classification problems. A binary classifier is then trained on each binary classification problem and predictions are made using the model that is the most confident.

For example, given a multi-class classification problem with examples for each class 'red,' 'blue,' and 'green'. This could be divided into three binary classification datasets as follows:

1. Binary Classification Problem 1: red vs [blue, green]
2. Binary Classification Problem 2: blue vs [red, green]
3. Binary Classification Problem 3: green vs [red, blue]

A possible downside of this approach is that it requires one model to be created for each class. For example, three classes requires three models. This could be an issue for large datasets (e.g. millions of rows), slow models (e.g. neural networks), or very large numbers of classes (e.g. hundreds of classes).

One-Vs-One for Multi-Class Classification

Unlike one-vs-rest that splits it into one binary dataset for each class, the one-vs-one approach splits the dataset into one dataset for each class versus every other class.

For example, consider a multi-class classification problem with four classes: 'red,' 'blue,' and 'green,' 'yellow.' This could be divided into six binary classification datasets as follows:

1. Binary Classification Problem 1: red vs. blue
2. Binary Classification Problem 2: red vs. green
3. Binary Classification Problem 3: red vs. yellow
4. Binary Classification Problem 4: blue vs. green
5. Binary Classification Problem 5: blue vs. yellow
6. Binary Classification Problem 6: green vs. yellow

This is significantly more datasets, and in turn, models than the one-vs-rest strategy described in the previous section.

The formula for calculating the number of binary datasets, and in turn, models, is as follows:

$$(NumClasses * (NumClasses–1))/2$$

We can see that for four classes, this gives us the expected value of six binary classification problems:

$$(NumClasses * (NumClasses–1))/2$$

$$4 * (4–1))/2 = 6$$

Each binary classification model may predict one class label and the model with the most predictions or votes is predicted by the one-vs-one strategy.

10 Advantages and disadvantages

- Logistic regression is relatively fast compared to other supervised classification techniques such as kernel SVM or ensemble methods
- It is far too simplistic for complex relationships between variables.
- logistic regression tends to underperform when the decision boundary is nonlinear.

11 Implementation

notebook demonstrating implementation of logistic regression : https://github.com/SSaishruthi/LogisticRegress

you tube tutorial : https://www.youtube.com/watch?v=VCJdg7YBbAQ

References

[1] Lecture 7 class recording: https://iiit-ac-in.zoom.us/rec/share/jP21R6dDFbByC2QQ6qxq2IpPq9X15_FSDjFv7SXEd Passcode: aM&HVb=5

[2] Towards data science: https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc

[3] Science Direct: https://www.sciencedirect.com/topics/medicine-and-dentistry/logistic-regression-analysis

[4] Wikipedia : https://en.wikipedia.org/wiki/Logistic_regression

[5] machinelearningmastery.com : https://machinelearningmastery.com/logistic-regression-for-machine-learning/

[6] https://web.stanford.edu/class/archive/cs/cs109/cs109.1178/lectureHandouts/220-logistic-regression.pdf

[7] https://towardsdatascience.com/probability-concepts-explained-maximum-likelihood-estimation-c7b4342fdbb1

[8] https://stats.stackexchange.com/questions/172900/can-gradient-descent-be-applied-to-non-convex-functions

[9] https://math.stackexchange.com/questions/3325382/how-to-check-if-a-function-is-convex

[10] https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/