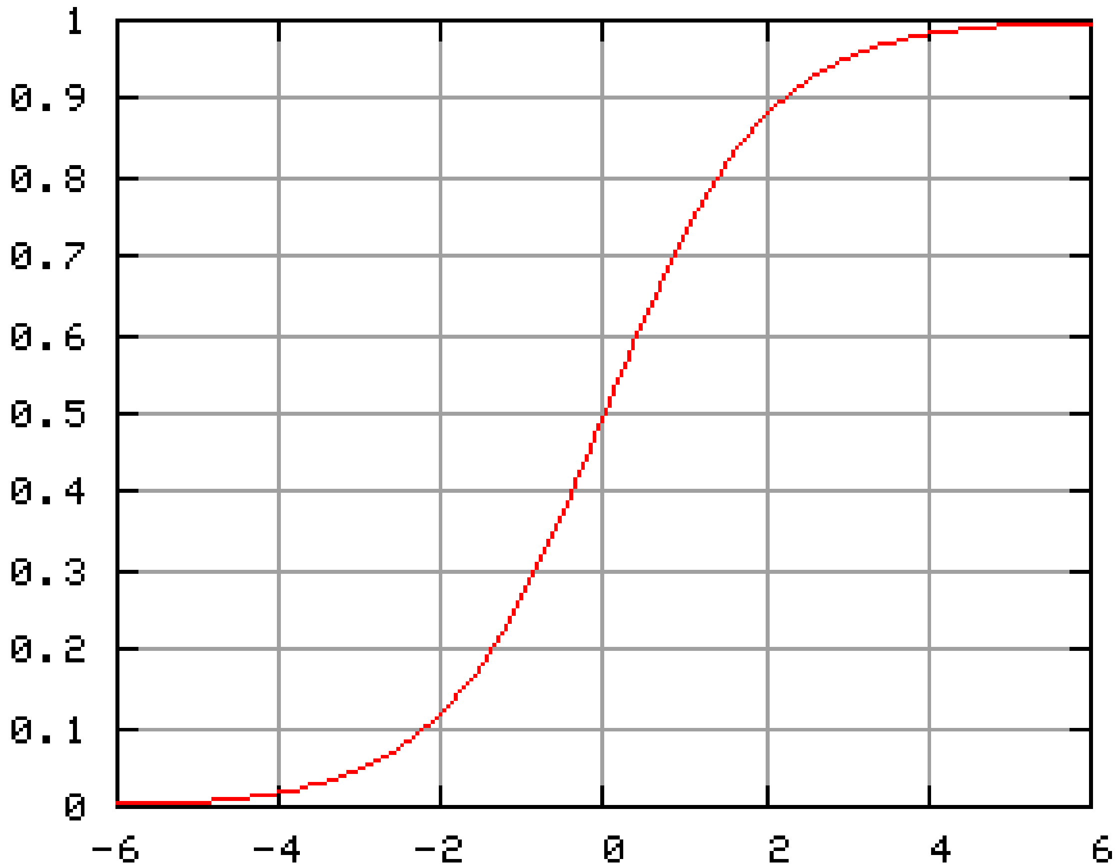
Logistic Regression

By
Muhammad Atif Tahir

Introduction

- Logistic regression is a form of regression analysis in which the outcome variable is binary
- What is the "Logistic" component?
- Instead of modeling the outcome, Y, directly, the method models the log odds(Y) using the logistic function

$$LOGIT(p) = \ln\left(\frac{p}{(1-p)}\right) = z \quad \Leftrightarrow \quad p = \frac{\exp(z)}{1+\exp(z)}$$



Introduction

- What is the “Regression” component?

- Methods used to quantify association between an outcome and predictor variables. Could be used to build predictive models as a function of predictors

The Logistic Regression Model

Logistic Regression:

$$\ln\left(\frac{\mathrm{P}(\mathrm{Y})}{1\text{-}\mathrm{P}(\mathrm{Y})}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{\mathrm{K}} X_{\mathrm{K}}$$

Linear Regression:

$$\mathrm{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{\mathrm{K}} X_{\mathrm{K}} + \varepsilon$$

The Logistic Regression Model

predictor variables

$$\ln\left(\frac{P(Y)}{1\text{-}P(Y)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_K X_K$$

dichotomous outcome

$$\ln\left(\frac{P(Y)}{1-P(Y)}\right)$$ is the log(odds) of the outcome.

The Logistic Regression Model

$$\ln\left(\frac{P(Y)}{1\text{-}P(Y)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_K X_K$$

intercept **model coefficients**

$$\ln\left(\frac{P(Y)}{1-P(Y)}\right)$$ is the log(odds) of the outcome.

Form for Predicted Probabilities

$$\ln\left(\frac{\mathrm{P}(\mathrm{Y})}{1\text{-}\mathrm{P}(\mathrm{Y})}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{\mathrm{K}} X_{\mathrm{K}}$$

$$\Updownarrow$$

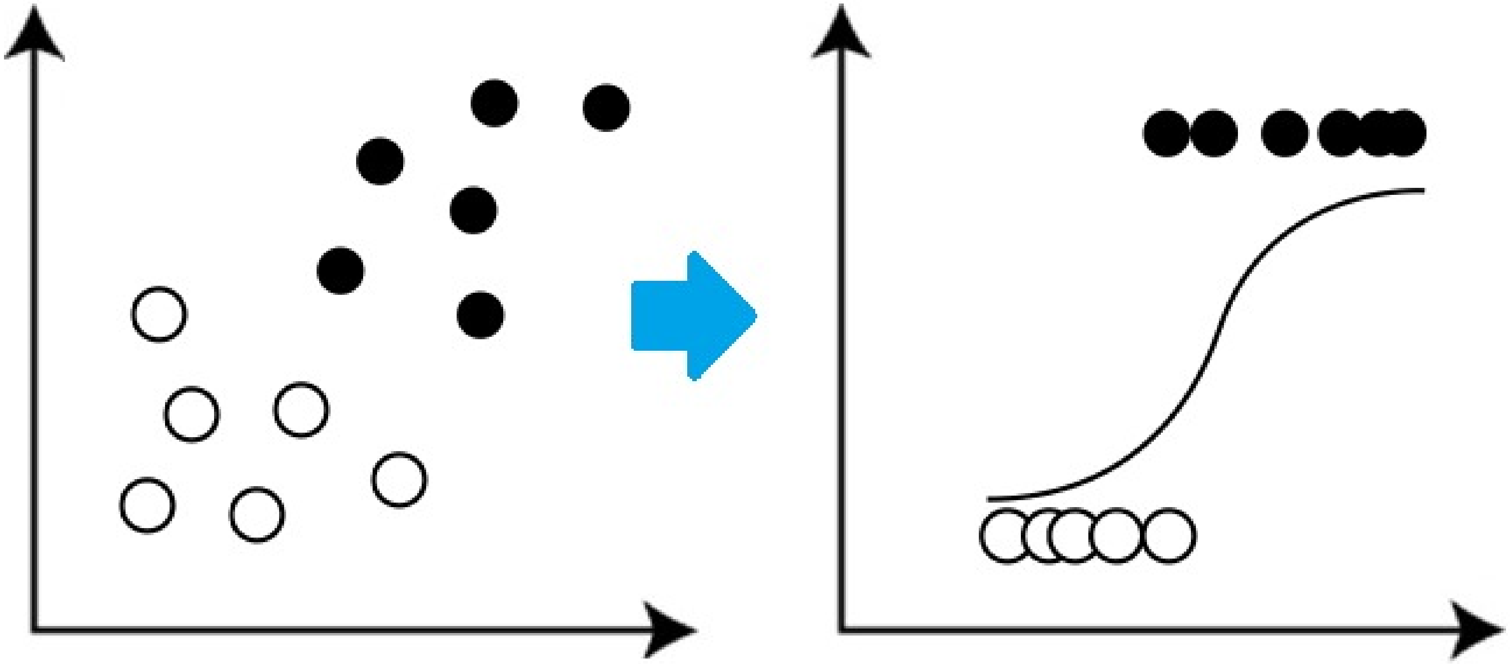
$$\mathrm{P}(\mathrm{Y}) = \frac{\exp\left(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{\mathrm{K}} X_{\mathrm{K}}\right)}{1 + \exp\left(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{\mathrm{K}} X_{\mathrm{K}}\right)}$$

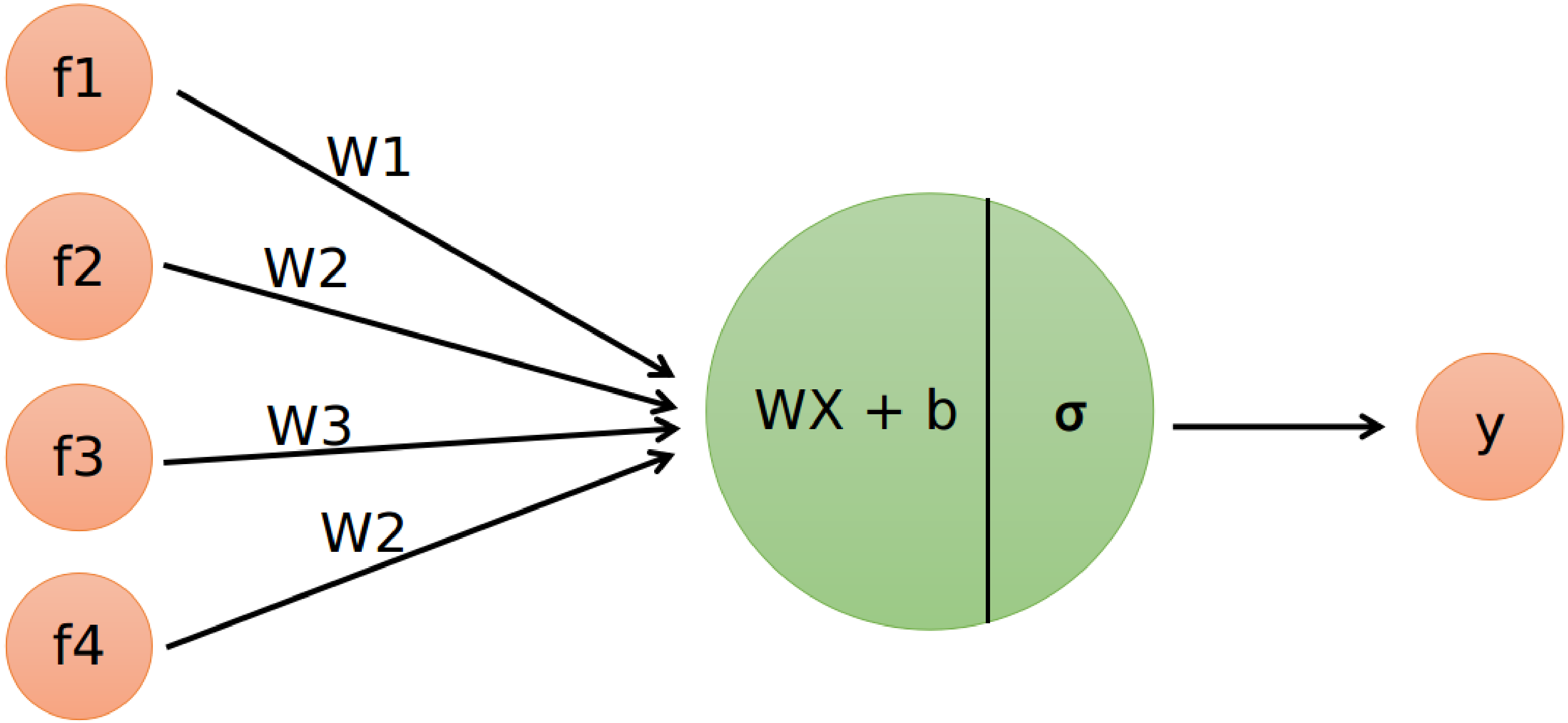
In this latter form, the logistic regression model directly relates the probability of Y to the predictor variables.

Commonality between linear and logistic regression

- Operating on the logit scale allows a linear model that is similar to linear regression to be applied
- Both linear and logistic regression are apart of the family of Generalized Linear Models (GLM)

LOGISTIC REGRESSION





Here, neuron has two operations: a linear part and activation function

Algorithm

- Calculate the prediction $\hat{y}$ using the current parameters (W and b)
- Calculate the loss of the current values
- Calculate the gradients of the loss function with respect to the parameters
- Adjust the weights (optimize) using the gradients
- Repeat for the number of epochs, i.e. the number of times to go through the provided examples (dataset)

Logistic Regression

$$z = b + a_1x_1 + a_2x_2 + a_3x_3$$
$$p = 1.0 \;/\; (1.0 + e^{-z})$$

Ex:

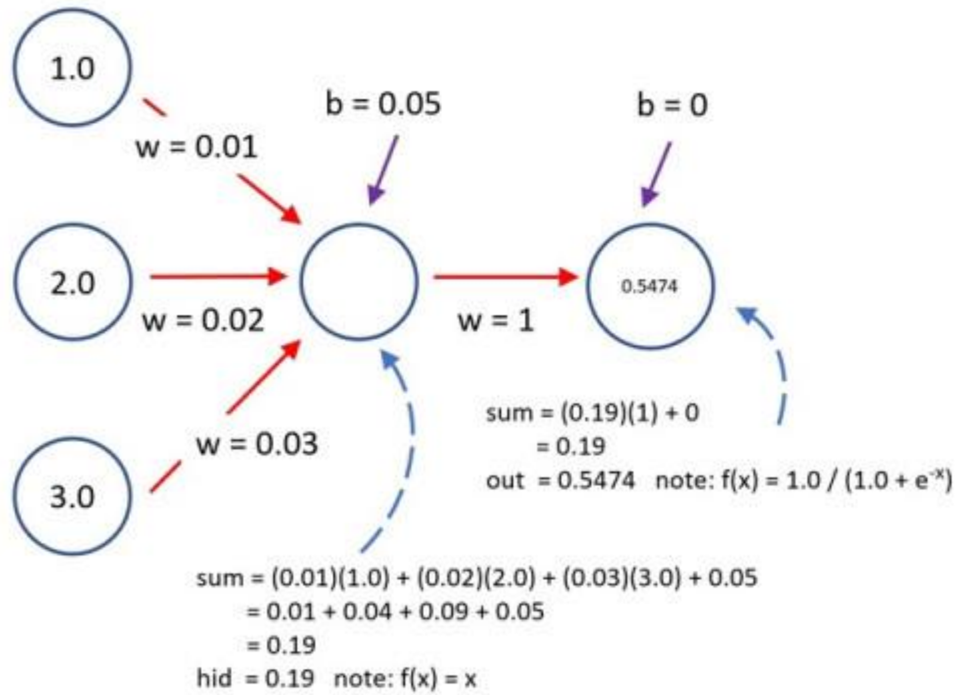
$x_1 = 1.0 \quad a_1 = 0.01$
$x_2 = 2.0 \quad a_2 = 0.02$
$x_3 = 3.0 \quad a_3 = 0.03$
$\qquad\qquad\quad b = 0.05$

$$\begin{aligned} z &= (0.05) + (0.01)(1.0) + (0.02)(2.0) + (0.03)(3.0) \\ &= 0.05 + 0.01 + 0.04 + 0.09 \\ &= 0.19 \end{aligned}$$

$$p = 1.0 \;/\; (1.0 + e^{-0.19})$$
$= 0.5474$ (predicted class = 1)

Neural Network

single hidden layer, identity activation $f(x) = x$
single output node, logistic sigmoid activation $f(x) = 1 / (1 + e^{-x})$



Loss Function

- Loss function is a function that one need to define to measure how good our predicted output is when the true label is y
- As square error seems like it might be a reasonable choice except that it makes gradient descent not work well
- it would be of no use as it would end up being a non-convex function with many local minimums
- it would be very difficult to minimize the cost value and find the global minimum

Loss Function

- So in logistic regression, we will actually define a different loss function that plays a similar role as squared error, that will give us an optimization problem that is convex

$$\text{Recap: } \hat{y} = \sigma(w^T x + b), \ \sigma(z) = \frac{1}{1+e^{-z}}$$

$$J(w,b) = \frac{1}{m}\sum^{m}\mathcal{L}(\hat{y}^{(i)}, y^{(i)})$$

$$= -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$

Loss Function

If $\quad y = 1: \qquad p(y|x) = \hat{y}$

If $\quad y = 0: \qquad p(y|x) = 1 - \hat{y}$

$P(y|x) = \hat{y}^{y}\,(1-\hat{y})^{(1-y)}$

If y = 1 => $\hat{y}$

If y = 0 => 1-$\hat{y}$

Loss Function

$$P(y|x) = \hat{y}^{y}\,(1\text{-}\hat{y})^{(1\text{-}y)}$$

$$\log(P(y|x)) = \log \hat{y}^{y}\,(1\text{-}\hat{y})^{(1\text{-}y)}$$

$$= \text{y}\log \hat{y} * (1\text{-}y)(1\text{-}\hat{y})$$

Above Equation describes a log likelihood that should be maximized. In order to turn this into loss function (something that we need to minimize), we'll just flip the sign of the above equation

Result in the cross entropy loss

Loss Function

$$L_{CE}(\hat{y}, y) = -\log p(y|x)$$

$$= \ -[y \log \hat{y} + (1-y)\log(1-\hat{y})]$$

$$L_{CE}(w,b) \ = \ -[y \log \sigma(w \cdot x + b) + (1-y)\log(1-\sigma(w \cdot x + b))]$$

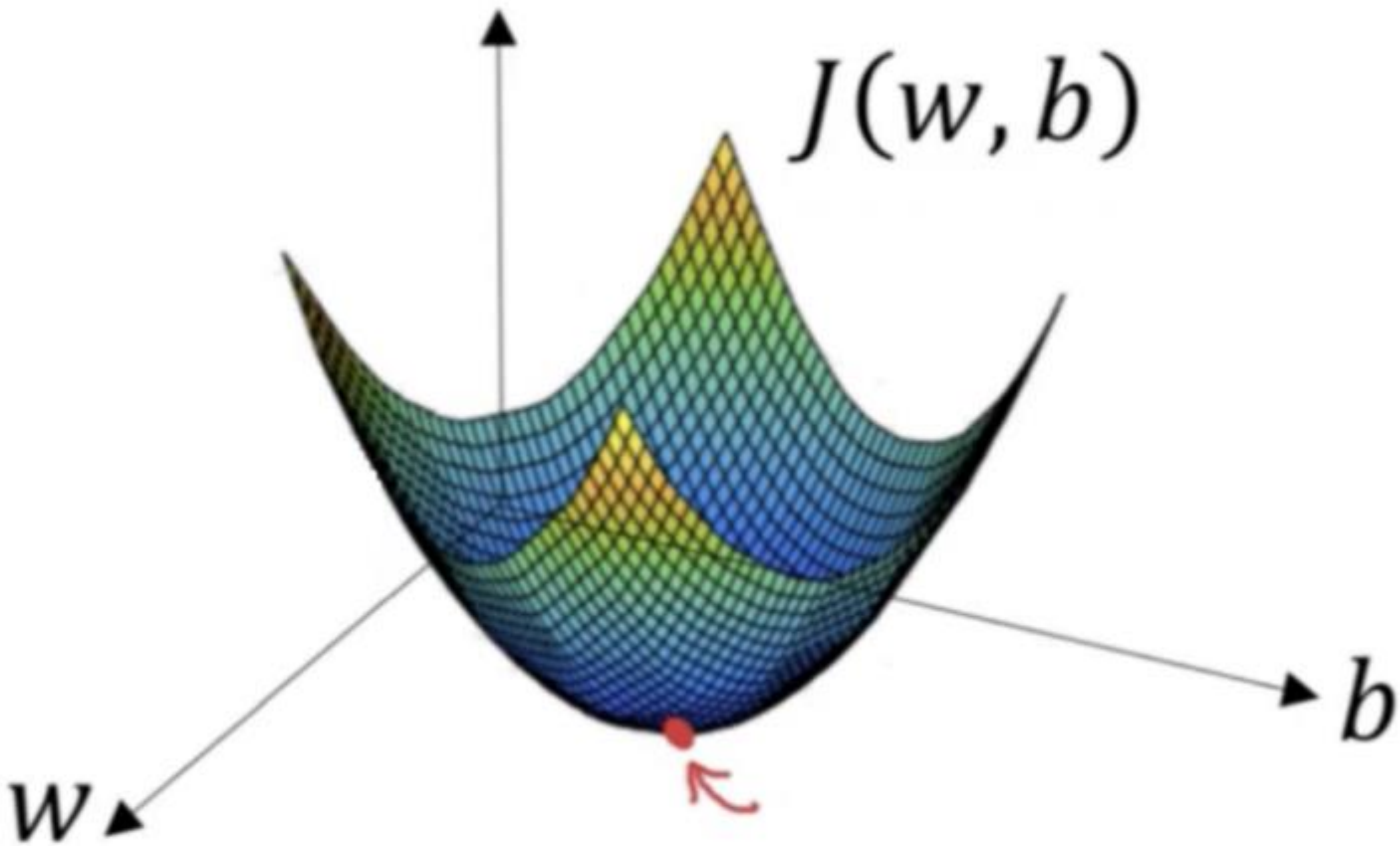
Gradient Descent

$$J(w,b) = \frac{\sum_1^m L(\hat{y}^i, y^i)}{m} = \frac{\sum_1^m y^i log\hat{y}^i + (1-y^i)log(1-\hat{y}^i)}{m}$$

In order to minimize the cost function for minimal error across the training data set to find w and b

The value of the parameters can be achieved using gradient descent technique.

Gradient Descent



More details about Logistic Regression

- Read Book Chapter 5

- Speech and Language Processing. Daniel Jurafsky & James H. Martin, 2019

References

- http://phs.wakehealth.edu/
- https://towardsdatascience.com/a-logistic-regression-from-scratch-3824468b1f88
- https://jamesmccaffrey.wordpress.com/2018/07/07/why-a-neural-network-is-always-better-than-logistic-regression/#jp-carousel-8874
- Notes from Coursera (Andrew Ng)
- https://medium.com/technology-nineleaps/logistic-regression-gradient-descent-optimization-part-1-ed320325a67e

thank you