1 Definitions For Real!

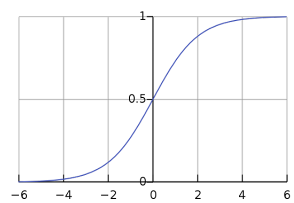
1. **Linear Regression**: Linear regression is a machine learning algorithm to perform the task of regression. Consider a dataset $\mathcal{D} = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), ..., (\mathbf{x}^{(N)}, y^{(N)})\}$, where $\mathbf{x}^{(i)} \in \mathbb{R}^M$ and $y^{(i)} \in \mathbb{R}$. We'll assume that the input $\mathbf{x}$ has already been augmented to include an extra 1 to allow for a bias term in $\boldsymbol{\theta}$. The linear regression hypothesis function is defined as:

$$\hat{y}^{(i)} = h_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}\right) = \boldsymbol{\theta}^\top \mathbf{x}^{(i)}$$

   where $\boldsymbol{\theta} \in \mathbb{R}^M$.

2. **Logistic Function**: The logistic function is part of a more general class of sigmoid functions characterized by an S-shaped curve. The logistic function is often useful for machine learning since we are required to differentiate functions and find their gradient.

$$g_{logistic}(z) = \frac{1}{1 + e^{-z}}$$



3. **Logistic Regression**: Logistic regression is used for classification tasks, but instead of predicting a specific class, it returns a real value that is meant to model the probability of the data point belonging to that class. As such, it assumes the following functional form for $P(Y = 1 \mid \mathbf{x}; \boldsymbol{\theta})$:

$$\hat{y}^{(i)} = h_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}\right) = P(Y = 1 \mid \mathbf{x}^{(i)}; \boldsymbol{\theta}) = \frac{1}{1 + e^{\boldsymbol{\theta}^\top \mathbf{x}^{(i)}}}$$

   Here we're starting to use a bit of probability notation, where capital letters represent random variables. In this case, Y is a binary random variable.

4. **One-hot Encoding**: A vector representation of a scalar integer n. Typically used to represent particular classes in a vector form, such that if there are a total of K classes, then the one-hot encoding of n would result in vector $\mathbf{u}$ where $u_n = 1$ and $u_i = 0 \;\forall\; i \neq n$.
   Ex. $K = 5$, $n = 3$ then $\mathbf{u} = [0, 0, 1, 0, 0]^\top$

5. **Multi-class Logistic Regression**: In class, we saw how to use logistic regression to model a binary variable. However, logistic regression can be used to learn a classifier for K classes too. There are 2 ways to implement this using logistic regression.

   In a K-class classification scenario, we can have the dataset $\mathcal{D} = \{(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{x}^{(2)}, \mathbf{y}^{(2)}), ..., (\mathbf{x}^{(N)}, \mathbf{y}^{(N)})\}$, where $\mathbf{x}^{(i)} \in \mathbb{R}^M$ and now $\mathbf{y}^{(i)} \in \{0,1\}^K$ is a one-hot vector filled with all zeros except a one in the k-th location when the data point belongs to class $k \in \{1, 2, \cdots, K\}$. Again, we'll assume that the input $\mathbf{x}$ has already been augmented to include an extra 1 to allow for a bias term in $\boldsymbol{\theta}$.

   (a) **One-vs-All**: Train K independent logistic regression models. Consists of the following two steps:

      i. Independently train K binary logistic regression models, one for each class. For each $1 \leq k \leq K$, treat samples of class k as positive and all other samples as negative. Then we perform binary logistic regression on this dataset, that is, find $P(Y_k = 1 \mid \mathbf{x}; \boldsymbol{\theta}_k)$

      ii. Perform majority vote on all $P(Y_k = 1 \mid \mathbf{x}; \boldsymbol{\theta}_k)$. That is, find $\hat{y} = \text{argmax}_k\, P(Y_k = 1 \mid \mathbf{x}; \boldsymbol{\theta}_k)$.

      Unfortunately, this one-vs-all approach 1) doesn't take advantage of the relationship between these classes and 2) loses the probabilistic result that we were interested in.

   (b) **Multi-class with Softmax**: Train a single model that considers all K classes all together. Each class will still have its own $\boldsymbol{\theta}_k$ but instead of one logistic function per class, we will tie all of the classes together using a softmax function. Consider K linear models, $z_k = \boldsymbol{\theta}_k^\top \mathbf{x}$. For a vector $\mathbf{z} = [z_1, z_2, \cdots, z_K]^\top \in \mathbb{R}^K$, the softmax function normalizes the input to output a vector of the same dimension:

$$g_{softmax}(\mathbf{z}) = \begin{bmatrix} e^{z_1} \\ e^{z_2} \\ \vdots \\ e^{z_K} \end{bmatrix} \frac{1}{\sum_{k=1}^K e^{z_k}}$$

      This guarantees that all entries in the softmax vector are in the range $(0, 1)$ and that the sum over all the elements in the softmax vector is 1.

      We can then use linear algebra to stack all K linear models together by creating one parameter matrix:

$$\Theta = \begin{bmatrix} - & \boldsymbol{\theta}_1^\top & - \\ - & \boldsymbol{\theta}_2^\top & - \\ & \vdots & \\ - & \boldsymbol{\theta}_K^\top & - \end{bmatrix} \qquad \text{and} \qquad \mathbf{z} = \Theta \mathbf{x}$$

$$\hat{\mathbf{y}} = h_\Theta(\mathbf{x}) = \begin{bmatrix} P(Y_1 = 1 \mid \mathbf{x}; \Theta) \\ P(Y_2 = 1 \mid \mathbf{x}; \Theta) \\ \vdots \\ P(Y_K = 1 \mid \mathbf{x}; \Theta) \end{bmatrix} = g_{softmax}(\Theta \mathbf{x})$$

   (c) **Cross-entropy Loss**: To compare probability distributions, we use the cross-entropy function, which can tell us how different two such distributions are. Consider a dataset of N samples, K classes, with the ith true and predicted assignment for class k as $y_k^{(i)}$ and $\hat{y}_k^{(i)}$ respectively.

$$J(\Theta) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} y_k^{(i)} \log(\hat{y}_k^{(i)})$$

2 Quick Logistic Regression Questions

2.1 What's the difference between linear regression and logistic regression?

List some differences between linear regression and logistic regression. In what situation would we use logistic regression instead of linear regression?

Linear regression assumes the data follows a linear function, while logistic regression models the data using a sigmoid function. We can also use logistic regression as a classification technique(when labels are binary), while we use linear regression when we are predicting some linear function on our data.

2.2 Just some logistics :)

Let $g(z) = g_{logistic}(z)$

1. We see that $g(z)$ falls strictly between (0,1). Given what we have discussed so far, what probability distribution does this graph represent?

$P(Y = 1 \mid x)$, where Y is a binary random variable representing the output class.

$P(Y = 1 \mid x) = \frac{1}{1+e^{-z}}$

and

$P(Y = 0 \mid x) = 1 - P(Y = 1 \mid x) = 1 - \frac{1}{1+e^{-z}} = \frac{e^{-z}}{1+e^{-z}}$

2. Now let's consider $\mathbf{x} \in \mathbb{R}^3$. For weight vector $\boldsymbol{\theta} = \begin{bmatrix} 1 \\ 4 \\ 3 \end{bmatrix}$

   (a) Define some $\mathbf{x}$ such that $\boldsymbol{\theta}^\top \mathbf{x} > 0$. What is the resulting $g\left(\boldsymbol{\theta}^\top \mathbf{x}\right)$?

   (b) Now define some $\mathbf{x}$ such that $\boldsymbol{\theta}^\top \mathbf{x} = 0$. What is the resulting $g\left(\boldsymbol{\theta}^\top \mathbf{x}\right)$?

Multiple correct $\mathbf{x}$. One example below

(a) Let $\mathbf{x} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$. Thus, $\boldsymbol{\theta}^\top \mathbf{x} = 8 > 0$ and $g(z) = \frac{1}{1+e^{-8}} = 0.99967$ which is close to 1

(b) Let $\mathbf{x} = \begin{bmatrix} 7 \\ -1 \\ -1 \end{bmatrix}$. Thus, $\boldsymbol{\theta}^\top \mathbf{x} = 0$ and $g(z) = \frac{1}{1+e^{-0}} = 0.5$

Explain the overall relationship between $g\left(\boldsymbol{\theta}^\top \mathbf{x}\right)$ and $\boldsymbol{\theta}^\top \mathbf{x}$.

Overall, we can see that the value of $g(z)$ depends on if z is greater than, less than, or equal to 0. If $z > 0$ then $g(z) > 0.5$ and if $z < 0$ then $g(z) < 0.5$. Finally if $z = 0$ then $g(z) = 0.5$. Thus we can see that based on the value of $g(z)$, we choose the appropriate binary class.

Since we have that $z = \boldsymbol{\theta}^\top \mathbf{x}$, we can say that $\boldsymbol{\theta}^\top \mathbf{x} = 0$ is our decision boundary.

3 Multiclass Logistic Regression Walkthrough

In a previous recitation, we saw how Pat loves to go on runs! Often times after his runs, he enjoys a good ice cream. We know that Pat has 3 favorite flavors and he only chooses from one of these: chocolate, vanilla, and strawberry. The ice cream he ends up choosing depends on two things: His mood ranges from 0 to 5 (sad to happy) and how hungry he is ranges from 0 to 5 (not at all hungry to very hungry). Since you are 10-315 students, the ice cream vendor reaches out to you for your help with predicting which ice cream flavor he would pick. Here is some information about the last 5 times Pat has had ice cream from the shop.

Mood (X_1)	Hunger (X_2)	Ice cream flavor (Y)
1	1	vanilla
4	5	strawberry
2	3	chocolate
3	4	chocolate
5	5	strawberry

Let's say your initial weight matrix Θ is defined as $\Theta = \begin{bmatrix} 0 & 3.8 & 3.9 \\ 0 & 4.6 & 3.8 \\ 0 & 5.4 & 3.1 \end{bmatrix}$ (the initial bias terms happen to be all zero).

As a first step, you map the possible flavors to the following class indices: {vanilla : 1, strawberry : 2, chocolate : 3}

1. Calculate the predicted softmax probabilities for each flavor for all training samples.

$\hat{\mathbf{y}}^{(i)} = g\left(\mathbf{z}^{(i)}\right)$ where $\mathbf{z}^{(i)} = \Theta \mathbf{x}^{(i)}$ and g is the softmax function.

$$\Theta\mathbf{x}^{(1)} = \begin{bmatrix} 7.7 \\ 8.4 \\ 8.5 \end{bmatrix} \quad \Theta\mathbf{x}^{(2)} = \begin{bmatrix} 34.7 \\ 37.4 \\ 37.1 \end{bmatrix} \quad \Theta\mathbf{x}^{(3)} = \begin{bmatrix} 19.3 \\ 20.6 \\ 20.1 \end{bmatrix} \quad \Theta\mathbf{x}^{(4)} = \begin{bmatrix} 27.0 \\ 29.0 \\ 28.6 \end{bmatrix} \quad \Theta\mathbf{x}^{(5)} = \begin{bmatrix} 38.5 \\ 42.0 \\ 42.5 \end{bmatrix}$$

$$\hat{\mathbf{y}}^{(1)} = \begin{bmatrix} 0.191 \\ 0.364 \\ 0.425 \end{bmatrix} \quad \hat{\mathbf{y}}^{(2)} = \begin{bmatrix} 0.037 \\ 0.553 \\ 0.410 \end{bmatrix} \quad \hat{\mathbf{y}}^{(3)} = \begin{bmatrix} 0.145 \\ 0.532 \\ 0.323 \end{bmatrix} \quad \hat{\mathbf{y}}^{(4)} = \begin{bmatrix} 0.075 \\ 0.554 \\ 0.371 \end{bmatrix} \quad \hat{\mathbf{y}}^{(5)} = \begin{bmatrix} 0.011 \\ 0.373 \\ 0.615 \end{bmatrix}$$

2. Create a corresponding one-hot vector, $\mathbf{y}^{(i)}$, output in the training set.

$$\mathbf{y}^{(1)} = \begin{bmatrix}1\\0\\0\end{bmatrix} \quad \mathbf{y}^{(2)} = \begin{bmatrix}0\\1\\0\end{bmatrix} \quad \mathbf{y}^{(3)} = \begin{bmatrix}0\\0\\1\end{bmatrix} \quad \mathbf{y}^{(4)} = \begin{bmatrix}0\\0\\1\end{bmatrix} \quad \mathbf{y}^{(5)} = \begin{bmatrix}0\\1\\0\end{bmatrix}$$

3. Compute the average cross-entropy loss $J(\Theta)$

$$J(\Theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} \log(\hat{y}_k^{(i)})$$

$$\begin{aligned} J(\Theta) = -\frac{1}{5}\big[&(\log 0.191 + 0 + 0) \\ &+ (0 + \log 0.553 + 0) \\ &+ (0 + 0 + \log 0.323) \\ &+ (0 + 0 + \log 0.371) \\ &+ (0 + \log 0.373 + 0)\big] \end{aligned}$$

$$J(\Theta) = 1.0711$$

4 K=2: Multi-class vs. Binary Logistic Regression

In the special case where $K = 2$, one can show that multi-class logistic regression reduces to binary logistic regression. This shows that multi-class logistic regression is a generalization of binary logistic regression.

1. Show that the following two equations are equivalent, where equation 1 is K=2 multi-class logistic regression and equation 2 is binary logistic regression:

$$P(Y_k = 1 \mid \mathbf{x}^{(i)}; \Theta) = \frac{\exp(\boldsymbol{\theta}_k^\top \mathbf{x}^{(i)})}{\sum_{l=1}^{K=2} \exp(\boldsymbol{\theta}_l^\top \mathbf{x}^{(i)})} \tag{1}$$

$$P(Y = k \mid \mathbf{x}^{(i)}; \theta_\alpha) = \begin{cases} \frac{1}{1+\exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))} & \text{if } k = 1 \\ \\ \frac{\exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))}{1+\exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))} & \text{if } k = 2 \end{cases} \tag{2}$$

We begin by simplifying equation 1 in terms of $k = 1$ and $k = 2$

$$\begin{bmatrix} P(Y_1 = 1 \mid \mathbf{x}^{(i)}; \Theta) \\ P(Y_2 = 1 \mid \mathbf{x}^{(i)}; \Theta) \end{bmatrix} = \frac{1}{\exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)}) + \exp(\boldsymbol{\theta}_2^\top \mathbf{x}^{(i)})} \begin{bmatrix} \exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)}) \\ \exp(\boldsymbol{\theta}_2^\top \mathbf{x}^{(i)}) \end{bmatrix}$$

$$\begin{aligned} P(Y_1 = 1 \mid \mathbf{x}^{(i)}; \Theta) &= \frac{\exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)})}{\exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)}) + \exp(\boldsymbol{\theta}_2^\top \mathbf{x}^{(i)})} \\ &= \frac{\exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)}) / \exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)})}{(\exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)}) + \exp(\boldsymbol{\theta}_2^\top \mathbf{x}^{(i)})) / \exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)})} \\ &= \frac{1}{1 + \exp(\boldsymbol{\theta}_2^\top \mathbf{x}^{(i)}) / \exp(\boldsymbol{\theta}_1^\top \mathbf{x}^{(i)})} \\ &= \frac{1}{1 + \exp((\boldsymbol{\theta}_2^\top - \boldsymbol{\theta}_1^\top)\mathbf{x}^{(i)})} \\ &= \frac{1}{1 + \exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))} \text{ , where } \boldsymbol{\theta}_\alpha = -(\boldsymbol{\theta}_2 - \boldsymbol{\theta}_1) \end{aligned}$$

$$\begin{aligned} P(Y_2 = 1 \mid \mathbf{x}^{(i)}; \Theta) &= 1 - P(Y_1 = 1 \mid \mathbf{x}^{(i)}; \Theta) \\ &= 1 - \frac{1}{1 + \exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))} \\ &= \frac{\exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))}{1 + \exp(-(\boldsymbol{\theta}_\alpha^\top \mathbf{x}^{(i)}))} \end{aligned}$$

The above two are of the same form as equation 2.