

# On logistic regression and maximum entropy approaches

Narayana Santhanam and Yixin Zhang  
 Department of Electrical and Computer Engineering  
 University of Hawaii  
 Honolulu, Hawaii  
 Email: {nsanathan,zhangyix}@hawaii.edu

**Abstract**—Logistic Regression is a widely used generalized linear model applied in classification settings to assign probabilities to class labels. It is also well known that logistic regression is a maximum entropy procedure subject to what are sometimes called the balance conditions. The dominant view in existing explanations are all discriminative, *i.e.*, modeling labels given the data. This paper adds to the maximum entropy interpretation, establishing a generative, maximum entropy explanation for the commonly used logistic regression training and optimization procedures. We show that logistic regression models the conditional distribution on the instance space given class labels with a maximum entropy model subject to a first moment constraint on the training data, and that the commonly used fitting procedure would be a Monte-Carlo fit for the generative view.

## I. INTRODUCTION

Logistic Regression, first introduced in [1], (see also [2]) is widely used [3]–[6] as a classification procedure both due to its ease of implementation, and its interpretability. This approach has been variously called as the multinomial logit model, softmax regression, and among natural language processing and machine learning practitioners, as the maximum entropy classifier [3], [7], see scikit-learn [8] implementation of logistic regression. Strictly speaking, logistic regression need not be a classification procedure alone. But since classification happens to be the most common use case, we focus on it.

However, The dominant way this classifier is introduced and understood is not from the maximum-entropy angle. Rather, a logit model is used as a link function in a generalized linear model [9], and further derivations flow naturally from it. Alternatively, it is also known that one could start with what are sometimes known as *balance equations* [7] that will be described below, and recover the logit model for class probabilities conditioned on input via a maximum entropy approach.

Both of these essentially are *discriminative* classification perspectives, where one models the conditional probability of labels given data but eschews modeling the data itself. For reasons we explain below, our paper also explores logistic regression from a maximum-entropy perspective, but we take a different direction from the above interpretations. By doing so, our main contribution is to show that it is also possible to recover the logistic regression approach from a *generative* perspective, where we model the conditional distribution of data given labels as a maximum entropy model.

### A. Problem setup

We first outline the broad strokes of the discriminative approach to position the questions we ask, and why we explore this well-known method in a new light.

Suppose  $\mathcal{X} \subset \mathbb{R}^d$  is an instance space, and  $\mathcal{Y}$  is a finite collection of labels (wolog, we will take  $\mathcal{Y} = \{0, \dots, k-1\}$ ). We denote  $X$  (resp  $Y$ ) to be random variables modeling the examples in  $\mathcal{X}$  (respectively labels from  $\mathcal{Y}$ ). We also assume in this paper that the probability model on  $\mathcal{X}$  is absolutely continuous with respect to the Lebesgue measure, and therefore, look for densities on  $\mathcal{X}$ .

In the discriminative view, one primarily considers the conditional distribution of  $Y$  given  $X$ , while the generative view models  $X$  and  $Y$  jointly. The training data will be *i.i.d.* samples from the joint distribution, denoted  $(X_i, Y_i)$  in our generative approach, and by  $(\mathbf{x}_i, Y_i)$  in the discriminative approach to emphasize that the distribution on  $X$  is not modeled.

The *multinomial logit model* assigns for  $\mathbf{x} \in \mathcal{X}$  and for  $j = 1, \dots, k-1$ ,

$$\frac{\mathbb{P}(Y = j|X = \mathbf{x})}{\mathbb{P}(Y = 0|X = \mathbf{x})} = \exp\left(\tilde{\beta}_{j0} + \tilde{\beta}_{j1}^T \mathbf{x}\right), \quad (1)$$

where  $\tilde{\beta}_{j0} \in \mathbb{R}$  and  $\tilde{\beta}_{j1} \in \mathbb{R}^d$  are parameters fitted from the training data. Then

$$\mathbb{P}(Y = 0|X = \mathbf{x}) = \frac{1}{1 + \sum_{j=1}^{k-1} \exp\left(\tilde{\beta}_{j0} + \tilde{\beta}_{j1}^T \mathbf{x}\right)}, \quad (2)$$

and from which  $\mathbb{P}(Y = j|X = \mathbf{x})$  for all  $1 \leq j \leq k-1$  is easily derived. We will reference the pair  $(\tilde{\beta}_{j0}, \tilde{\beta}_{j1})$  as  $\tilde{\beta}_j$  parameters, and the parameters  $\tilde{\beta}_j$ ,  $j = 0, \dots, k-1$  as the  $\beta$  parameters for simplicity.

Suppose the training data is  $(\mathbf{x}_i, Y_i)$ ,  $i = 1, \dots, n$ . The above expressions yield  $\mathbb{P}(Y_i|X = \mathbf{x}_i)$  in terms of the logit model parameters  $\tilde{\beta}_j$ ,  $j = 1, \dots, k-1$ . The parameters  $\tilde{\beta}_j$  are then calculated by maximizing the likelihood

$$\mathbb{P}(Y_1, \dots, Y_n | \mathbf{x}_1, \dots, \mathbf{x}_n) = \prod_i \mathbb{P}(Y_i | \mathbf{x}_i).$$

In practice, it is common to add to the log likelihood, an  $\ell_2$  or  $\ell_1$  penalty on the  $\tilde{\beta}_j$  parameters.

One can get to the same optimization from a different perspective. Suppose we did not start from any explicit form

for the conditional probabilities of  $\mathbb{P}(Y|X = \mathbf{x}_i)$ . We would recover the logistic regression approach, specifically the logit models for  $\mathbb{P}(Y|X = \mathbf{x}_i)$ ,  $i = 1, \dots, n$ , when we search for the maximum entropy model subject to the *balance* constraints for  $j = 0, \dots, k-1$ ,

$$\begin{aligned} \sum_{i=1}^n \mathbf{1}(Y_i = j) \mathbf{x}_i &= \sum_{l=1}^n \mathbb{P}(Y = j | X = \mathbf{x}_l) \mathbf{x}_l \\ \sum_{i=1}^n \mathbf{1}(Y_i = j) 1 &= \sum_{l=1}^n \mathbb{P}(Y = j | X = \mathbf{x}_l), \end{aligned} \quad (3)$$

where  $\mathbf{1}(\cdot)$  is the indicator function (value 1 when the argument is true, 0 else). These balance conditions happen to be a restatement of the gradient being zero when maximizing the likelihood in the generalized linear models (GLM) approach mentioned above.

The above insight, while illuminating, still seems unsatisfactory. The balance conditions did come up in the GLM approach after all. Rather than pull the logit model out of a magic hat, doesn't the max-entropy explanation just pull the balance condition out of it?

Furthermore, strictly speaking, this maximum entropy view only yields  $\mathbb{P}(Y|X = \mathbf{x}_i)$ , *i.e.*, the conditional distribution of the labels given the training data points. While it may seem natural to extend the same functional form to all  $\mathbf{x} \in \mathcal{X}$ , the question remains as to the formal justification for it. There can conceivably be other ways to generalize to other instances  $\mathbf{x} \in \mathcal{X}$ —for example, using nearest-neighbor like approaches using the  $n$  conditional distributions from the training points. Most importantly, why should we not say anything about the conditional distribution of  $X$  given a class label?

This paper tries to dig deeper in this direction. We construct a generative view of logistic regression which models the distribution on the instance space  $X$  given each class label by a maximum entropy model subject to a first moment constraint (on  $\mathbb{E}[X|Y = j]$ ). We show that the standard fitting procedure used in the discriminative approach is a Monte-Carlo approximation of fitting the training data in the generative view. In the limit of large training data, the law of large numbers guarantees consistency of the approximation.

## II. BACKGROUND

### A. Maximum entropy approach

The maximum entropy principle is one way of formalizing the notion of making as few additional assumptions as possible, given a set of constraints. Suppose  $X$  is a random variable taking values in the set  $\mathcal{X}$ .

Given  $d$  different constraints on a random variable  $X$ ,  $\mathbb{E}r_i(X) = \alpha_i$  where for  $1 \leq i \leq d$ ,  $r_i : \mathcal{X} \rightarrow \mathbb{R}$  are real valued functions and  $\alpha_i \in \mathbb{R}$ , the distribution on  $X$  with maximum entropy is [10] (if  $X$  is discrete,  $f$  below is a probability mass function, and if  $X$  is continuous,  $f$  is a probability density function):

$$f(x) = \exp \left( \beta_0 + \sum_{i=1}^d \beta_i r_i(x) \right).$$

In the above,  $\beta_0, \dots, \beta_d$  are numbers chosen so that  $f$  is a probability mass or density function (*i.e.*, integrates/sums to 1) and  $\mathbb{E}r_i(X) = \alpha_i$  for  $i = 1, \dots, d$ .

### B. Discriminative perspective of logistic regression

The classical discriminative presentation starts from (1), yielding for  $j = 1, \dots, k-1$ ,

$$\mathbb{P}(Y = j | X = \mathbf{x}) = \frac{\exp(\tilde{\beta}_{j0} + \tilde{\beta}_j^T \mathbf{x})}{1 + \sum_{i=1}^{k-1} \exp(\tilde{\beta}_{i0} + \tilde{\beta}_i^T \mathbf{x})}$$

as well as Equation (2). Note that there are  $k-1$  sets of parameters  $\tilde{\beta}_i$ , for  $i = 1, \dots, k-1$ .

To simplify exposition, we will consider the binary case  $k = 2$ , so there is only one set of parameters, which we denote as  $\tilde{\beta}$ . We can write

$$\mathbb{P}(Y = y | X = \mathbf{x}) = \frac{\exp(y\tilde{\beta}_0 + y\tilde{\beta}^T \mathbf{x})}{\sum_{\tilde{y}=0}^1 \exp(\tilde{y}\tilde{\beta}_0 + \tilde{y}\tilde{\beta}^T \mathbf{x})}.$$

We are given training data  $(\mathbf{x}_i, Y_i)$ ,  $i = 1, \dots, n$ . To fit the model above to the training data, we write the likelihood of training labels given examples, assuming that the label  $Y_i$  given the instance  $\mathbf{x}_i$  is independent of all other labels and does not depend on other instances, to yield

$$\begin{aligned} \mathbb{P}(Y_1, \dots, Y_n | \mathbf{x}_1, \dots, \mathbf{x}_n) &= \prod_{i=1}^n \mathbb{P}(Y_i | \mathbf{x}_i) \\ &= \frac{\prod_{i=1}^n \exp(Y_i \tilde{\beta}_0 + Y_i \tilde{\beta}^T \mathbf{x}_i)}{\left( \sum_{\tilde{y}=0}^1 \exp(\tilde{y} \tilde{\beta}_0 + \tilde{y} \tilde{\beta}^T \mathbf{x}_i) \right)^n}. \end{aligned}$$

It is easier to work with log-likelihood,

$$\log \prod \mathbb{P}(Y_i | \mathbf{x}_i) = \sum_i^n \log \mathbb{P}(Y_i | \mathbf{x}_i).$$

We just find the value of  $\tilde{\beta}$  that maximizes the above log likelihood. It is easy to verify that these are exactly those satisfying (3) for  $j = 0, 1$ .

To see this, first note that

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

and

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

so setting the gradient to 0, we get

$$\begin{aligned} 0 &= \nabla_{\tilde{\beta}} \log \prod \mathbb{P}(Y_i | \mathbf{x}_i) = \sum_{i=1}^n \nabla_{\tilde{\beta}} \log \mathbb{P}(Y_i | \mathbf{x}_i) \\ &= \sum_i \frac{\mathbf{1}(Y_i = 1)}{\mathbb{P}(1 | \mathbf{x}_i)} \mathbb{P}(1 | \mathbf{x}_i) (1 - \mathbb{P}(1 | \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix} \\ &\quad - \sum_i \frac{\mathbf{1}(Y_i = 0)}{\mathbb{P}(0 | \mathbf{x}_i)} \mathbb{P}(0 | \mathbf{x}_i) (1 - \mathbb{P}(0 | \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix} \end{aligned}$$

The above equations are easily rewritten to yield Equations 3 again.

At this point, the parameters  $\tilde{\beta}$  satisfying the above equations are obtained iteratively by Newton-Raphson's method.

a) *Note:* : When solving for the parameters, it is often better to add in a  $\ell_2$  regularization on  $\tilde{\beta}$ s. To see why, note that if the data is linearly separable, there exists parameter values  $\tilde{\beta}$  such that ensures  $\mathbb{P}(Y_i|\mathbf{x}_i) \geq \mathbb{P}(1-Y_i|\mathbf{x}_i)$  for all  $i$ . If this is the case, scaling  $\tilde{\beta}$  by any number greater than 1 increases the likelihood, hence pushing the optimal parameters to infinity, a meaningless exercise. In these cases, it is useful to limit the length of  $\tilde{\beta}$ , thus implementations (including scikit-learn) often use  $\ell_2$  regularization by default.

### C. Discriminative max-ent explanation

In natural language processing, it is common to start with (3) as a set of given constraints. One then searches for  $\mathbb{P}(Y|\mathbf{x}_i)$ ,  $i = 1, \dots, n$  using the maximum entropy formulation, which then recovers (1) as its solution. But as explained in the introduction, we search for a different explanation since this (i) does not answer the question of why (3) is sacrosanct, and (ii) why the solution for conditional distribution on labels for the training examples must generalize in the same functional form everywhere when other natural approaches also exist.

## III. GENERATIVE MAX-ENT MODELING

We model the joint probability model on the examples  $X$  taking values in  $\mathcal{X}$  and labels  $Y$  taking values in  $\mathcal{Y}$ . The training data is assumed to be *i.i.d.* copies of  $(X, Y)$ , denoted by  $(X_i, Y_i)$ ,  $i = 1, \dots, n$ .

### A. Maximum entropy model for $X$ given $Y$

The first step in the generative approach is natural given the form of logistic regression. We build a model for each class,  $f(X|Y = j)$ ,  $j = 1, \dots, k$ , using the maximum entropy model when we constrain  $\mathbb{E}[X|Y = j]$ . Note that  $X \in \mathcal{X}$  is a vector in  $\mathbb{R}^d$  (*i.e.*, has  $d$  components), so constraining  $\mathbb{E}[X|Y = j]$  corresponds to  $d$  moment constraints (one for each component of the vector).

We begin with a technical claim.

**Claim 1.** Let  $m \geq 1$  and  $g : \mathbb{R}^d \rightarrow \mathbb{R}^m$ . An unbiased estimate of  $\mathbb{E}[g(X)|Y = j]$  from independently generated training data  $(X_i, Y_i)$  is

$$\frac{\sum_{i=1}^n g(X_i)\mathbf{1}(Y_i = j)}{N_j},$$

where  $N_j = \sum_i \mathbf{1}(Y_i = j)$  is the number of training examples that have  $j$  as their label.

**Proof** For any number  $1 \leq r \leq n$ , let  $A_r \subset \{0, 1\}^n$  be the set of binary strings with exactly  $r$  1s. For all  $\mathbf{a} \in A_r$ , let

$$\mathcal{Y}_{\mathbf{a}} = \{(y_i)_{1 \leq i \leq n} \in \mathcal{Y}^n : \text{for all } i, y_i = j \text{ iff } a_i = 1\}$$

be the event that the training labels are  $j$  in exactly the positions marked by the string  $\mathbf{a}$ . Let  $\mathcal{Y}_r = \cup_{\mathbf{a} \in A_r} \mathcal{Y}_{\mathbf{a}}$  be the

event that there are exactly  $r$  training labels that are equal to  $j$ .

As the examples are all independent, we also have for all  $0 \leq r \leq n$ ,  $\mathbf{a} \in A_r$  and  $Y_1^n \in \mathcal{Y}_{\mathbf{a}}$  that

$$f(X_1, \dots, X_n | Y_1, \dots, Y_n, \mathcal{Y}_{\mathbf{a}}, \mathcal{Y}_r) = f(X_1, \dots, X_n | Y_1, \dots, Y_n).$$

Therefore, allowing  $R$  to be a Binomial( $n, \mathbb{P}(Y = j)$ ) random variable, and  $\mathbf{a}$  to be uniformly distributed in  $A_R$ ,

$$\begin{aligned} & \mathbb{E} \left[ \frac{\sum_{i=1}^n g(X_i)\mathbf{1}(Y_i = j)}{N_j} \right] \\ &= \mathbb{E} \left[ \mathbb{E} \left[ \frac{\sum_{i=1}^n g(X_i)\mathbf{1}(Y_i = j)}{R} \middle| \mathcal{Y}_{\mathbf{a}}, \mathcal{Y}_R, \mathbf{a}, R \right] \right] \\ &= \mathbb{E} \left[ R \frac{\mathbb{E}[g(X)|Y = j]}{R} \middle| R \right] \\ &= \mathbb{E}[g(X)|Y = j] \quad \square \end{aligned}$$

**Problem 1.** Given training data  $(X_i, Y_i)$ ,  $i = 1, \dots, n$ , for all  $j \in \mathcal{Y}$ , find the pdf  $f(X|Y = j)$  with maximum entropy among those that satisfy

$$\mathbb{E}[X|Y = j] = \frac{\sum_{i=1}^m X_i}{N_j}, \quad (4)$$

where  $N_j$  is the number of labels  $Y_i$  that equal  $j$ .

**Approach** The maximum entropy model, from the prior section, is

$$f(X = \mathbf{x} | Y = j) = \exp(\beta_{j0} + \beta_j^T \mathbf{x}),$$

where  $\beta_{j0} \in \mathbb{R}$ ,  $\beta_j \in \mathbb{R}^d$  has the same number of coordinates as the training instance  $\mathbf{x}$ , with  $\beta_{j0}$  and  $\beta_j$  chosen to satisfy

- $\int_{\mathbf{x} \in \mathcal{X}} df(X|Y = j) = 1$
- the constraints in Equation (4)

Obtaining  $\beta_j$  from the above equations is non-trivial in general, given the integration should be over  $\mathcal{X}$ , the space of all instances. In practical examples, the exact domain  $\mathcal{X}$  of valid examples is very difficult or impossible to pin down (though the use of  $\mathcal{X}$  for description and abstraction purposes is ubiquitous).

Notwithstanding the computation of  $\beta$ s to satisfy the above constraints, if we only focus on the form of  $\mathbb{P}(Y = j|X = \mathbf{x})$ , the generative model obviously mimics that of the discriminative approach barring cosmetic differences. Indeed, taking class 0 as a reference, and we have for  $j = 1, \dots, k-1$ , using Bayes Rule

$$\frac{f(X = \mathbf{x} | Y = j)}{f(X = \mathbf{x} | Y = 0)} = \frac{\mathbb{P}(Y = j|X = \mathbf{x})}{\mathbb{P}(Y = 0|X = \mathbf{x})} \cdot \frac{\mathbb{P}(Y = 0)}{\mathbb{P}(Y = j)}.$$

Rearranging, we get

$$\frac{\mathbb{P}(Y = j|X = \mathbf{x})}{\mathbb{P}(Y = 0|X = \mathbf{x})} = \exp(\tilde{\beta}_{j0} + \tilde{\beta}_j^T \mathbf{x}).$$

Here  $\tilde{\beta}_{j0} = \beta_{j0} - \beta_{00} + \ln \frac{\mathbb{P}(Y=j)}{\mathbb{P}(Y=0)}$ , and  $\tilde{\beta}_j = \beta_j - \beta_0$ , where  $\beta_{j0}$ ,  $\beta_j$  were the constants we used for maximum entropy modeling for classes  $j$ . Using the fact that  $\sum_j \mathbb{P}(Y = j|X =$

$\mathbf{x}) = 1$  for all  $\mathbf{x}$ , we recover Equation (2) and for all  $1 \leq j \leq k-1$ ,

$$\mathbb{P}(Y = j|X = \mathbf{x}) = \frac{\exp(\tilde{\beta}_{j0} + \tilde{\beta}_j^T \mathbf{x})}{1 + \sum_{i=1}^{k-1} \exp(\tilde{\beta}_{i0} + \tilde{\beta}_i^T \mathbf{x})}. \quad \square$$

### B. Solving for parameters

At this stage it is not clear how one could find the parameters  $\beta$  that satisfy (4) or at least, the  $\tilde{\beta}$  parameters (the  $\beta$  parameters shifted by that of the reference class). Indeed, the main challenge with formulating the generative approach is to show a way to compute the parameters.

We show an approximate way to compute the  $\tilde{\beta}$  parameters that satisfy the constraints of the generative maximum entropy modeling. Rather than compute the integral on the left side of (4), we look for its Monte Carlo approximations that can be obtained from the training data.

**Claim 2.** Let  $m \geq 1$  and let  $g: \mathbb{R}^d \rightarrow \mathbb{R}^m$ . Then

$$\mathbb{E}[g(X)|Y = j] = \frac{\mathbb{E}[g(X)\mathbb{P}(Y = j|X)]}{\mathbb{P}(Y = j)}.$$

**Proof** Note that

$$g(X)\mathbb{P}(Y = j|X) = g(X) \frac{f(X|Y = j)\mathbb{P}(Y = j)}{f(X)}.$$

Therefore the quantity  $g(X)\mathbb{P}(Y = j|X)$  can be thought of as a (vector-valued) random variable whose expectation equals  $\mathbb{E}[g(X)|Y = j]\mathbb{P}(Y = j)$ .  $\square$

Our training data contains  $n$  points,  $(X_i, Y_i)$ ,  $i = 1, \dots, n$ .

To enforce the constraints on  $\mathbb{E}[X|Y = j]$  in Equation (4),  $0 \leq j \leq k-1$ , we can think of

$$\frac{1}{n} \sum_i X_i \mathbb{P}(Y = j|X_i)$$

to be a Monte Carlo estimate of

$$E[X|Y = j]\mathbb{P}(Y = j).$$

Rewriting (4) as

$$\mathbb{E}[X|Y = j]\mathbb{P}(Y = j) = \frac{\sum_{i=1}^n X_i \mathbf{1}(Y_i = j)}{N_j} \mathbb{P}(Y = j),$$

where  $N_j$  is the number of labels  $Y_i$  that equal  $j$ . Replacing the left side by its Monte Carlo estimate from above, we get

$$\frac{1}{n} \sum_i X_i \mathbb{P}(Y = j|X_i) = \frac{\sum_{i=1}^n X_i \mathbf{1}(Y_i = j)}{N_j} \mathbb{P}(Y = j).$$

Using  $\mathbb{P}(Y = j) = \frac{N_j}{N}$  (the natural unbiased estimate), we recover

$$\sum_{i=1}^n X_i \mathbb{P}(Y = j|X_i) = \sum_l \mathbf{1}(Y_l = j) X_l,$$

the exact balance equations that arises/is used in the discriminative formulations of Logistic Regression.

To enforce  $\int_{\mathcal{X}} df(X|Y = j) = 1$ , we use a similar approach, this time requiring  $\mathbb{E}[\mathbf{1}|Y = j] = 1$  (writing

$g(X) = 1$  in Claims 1 and 2). Doing so yields the remaining balance equations,

$$\sum_{i=1}^n \mathbb{P}(Y = j|X_i) = \sum_l \mathbf{1}(Y_l = j).$$

To summarize, we used Maximum Entropy modeling of  $X$  given each class label, subject to first moment (expectation) constraints on  $X$  given the class label to obtain Equation (4). But rather than use an integral to compute  $\mathbb{E}[X|Y = j]$  to get the parameters, we obtain the parameters using Equation (3) instead, where  $\mathbb{E}[X|Y = j]$  is replaced by its Monte Carlo estimate.

## IV. DISCUSSION

While conventional logistic regression is usually used with the first moment constraints (on  $\mathbb{E}[X|Y = j]$ ), there is no real reason to stick to only these. Indeed, it may be desirable in several applications to use other constraints on the conditional models for the data given labels, for example, see [6]. For constraints  $\mathbb{E}[g(X)|Y = j] = \sum_{i:Y_i=j} g(X_i)$ , the balance equations would be

$$\sum_{i=1}^n g(\mathbf{x}_i) \mathbb{P}(Y = j|\mathbf{x}_i) = \sum_l g(\mathbf{x}_l) \mathbf{1}(Y_l = j)$$

$$\sum_{i=1}^n \mathbb{P}(Y = j|\mathbf{x}_i) = \sum_l \mathbf{1}(Y_l = j).$$

In this preliminary version, the full import and study of these techniques has not been carried out, and remains an open problem. Questions on convergence—how quickly the Monte Carlo approximations converge to true values, and how the approximation affects downstream tasks also remain open.

## ACKNOWLEDGMENT

The authors were supported by NSF grants NRT-AI 2244574 “Data in Engineering and Society: Converging Applications, Research, and Training Enhancements for Students” and CCF-2324396 “Theory for Learning Lossless and Lossy Coding”.

## REFERENCES

- [1] D. R. Cox, “The regression analysis of binary sequences,” *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 20, no. 2, pp. 215–232, 1958.
- [2] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, “The elements of statistical learning: data mining, inference and prediction,” *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005.
- [3] A. Berger, S. A. Della Pietra, and V. J. Della Pietra, “A maximum entropy approach to natural language processing,” *Computational linguistics*, vol. 22, no. 1, pp. 39–71, 1996.
- [4] V. Chandrasekaran, J. K. Johnson, and A. S. Willsky, “Maximum entropy relaxation for graphical model selection given inconsistent statistics,” in *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*. IEEE, 2007, pp. 625–629.
- [5] E. Scharfenaker and J. Yang, “Maximum entropy economics,” *The European Physical Journal Special Topics*, vol. 229, pp. 1577–1590, 2020.

- [6] R. R. Stein, D. S. Marks, and C. Sander, "Inferring pairwise interactions from biological data using maximum-entropy probability models," *PLoS computational biology*, vol. 11, no. 7, p. e1004182, 2015.
- [7] J. Mount, "The equivalence of logistic regression and maximum entropy models," 2011. [Online]. Available: <https://github.com/WinVector/Examples/blob/main/dfiles/LogisticRegressionMaxEnt.pdf>
- [8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [9] J. A. Nelder and R. W. Wedderburn, "Generalized linear models," *Journal of the Royal Statistical Society Series A: Statistics in Society*, vol. 135, no. 3, pp. 370–384, 1972.
- [10] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006.