American Journal of Theoretical and Applied Statistics
2021; 10(4): 184-193
http://www.sciencepublishinggroup.com/j/ajtas
doi: 10.11648/j.ajtas.20211004.12
ISSN: 2326-8999 (Print); ISSN: 2326-9006 (Online)

An Entropy Based Objective Bayesian Prior Distribution

Jamie Watson

AiZiA, Santa Barbara, the United States of America

Email address:
visionaryphysicist@outlook.com

To cite this article:
Jamie Watson. An Entropy Based Objective Bayesian Prior Distribution. *American Journal of Theoretical and Applied Statistics*. Vol. 10, No. 4, 2021, pp. 184-193. doi: 10.11648/j.ajtas.20211004.12

Received: July 20, 2021; **Accepted:** August 6, 2021; **Published:** August 23, 2021

Abstract: Bayesian Statistical Analysis requires that a prior probability distribution be assumed. This prior is used to describe the likelihood that a given probability distribution generated the sample data. When no information is provided about how data samples are drawn, a statistician must use what is called an, "objective prior distribution" for analysis. Some common objective prior distributions are the Jeffery's prior, Haldane prior, and reference prior. The choice of an objective prior has a strong effect on statistical inference, so it must be chosen with care. In this paper, a novel entropy based objective prior distribution is proposed. It is proven to be uniquely defined given a few postulates, which are based on well accepted properties of probability distributions. This novel objective prior distribution is shown to be the exponential of the entropy information in a probability distribution (e^S), which suggests a strong connection to information theory. This result confirms the maximal entropy principle, which paves the way for a more robust mathematical foundation for thermodynamics. It also suggests possible connection between quantum mechanics and information theory. The novel objective prior distribution is used to derive a new regularization technique that is shown to improve the accuracy of modern day artificial intelligence on a few real world data sets on most test runs. On just a couple of trials, the new regularization technique overly regularized a neural network and lead to poorer results. This showed that, while often quite effective, this new regularization technique must be used with care. It is anticipated that this novel objective prior will be an integral part of many new algorithms that focus on finding an appropriate model to describe a data set.

Keywords: Statistics, Data Science, Artificial Intelligence, Information Theory

1. Introduction

In 1763 Bayes discovered how to make the ideal probability distribution [1]. The idea behind his Bayesian inference work is to calculate the average loss between a statistician chosen probability distribution and all possible probability distributions weighted by both how likely they are to exist and how likely it is for them to have produced the sample data. The probability distribution that minimizes this loss is the ideal fit for the data. In order to use Bayes' formula, the sample data points, the loss function, and the prior probability distribution (which represents how likely a probability distribution is to exist) must all be known. In many cases the prior probability distribution is unknown, so it must be guessed in order to use Bayes' work. There have been many attempts to discover what the best prior distribution is for arbitrary sample points, often called an, "objective Bayesian prior distribution" [2–5], but there is no consensus among the statistics community that any given objective prior distribution is the correct one. In this paper, an objective Bayesian prior distribution is derived from simple postulates. The prior distribution is shown to be the exponential of the entropy (e^S) of the probability distribution. This result has a strong connection with information theory [6]. It will be shown to lead to a new regularization technique for neural networks [7] that can boost their accuracy under the right circumstances.

Section two contains the mathematical proof of the proposed prior distribution given two simple postulates. Section three shows how to account for this prior distribution in gradient decent algorithms often used in data science [8] and tests of the prior distribution using neural networks [9]. Section four is the conclusion of the paper.

2. Proof of Novel Objective Prior Distribution

This section provides a mathematical proof that an objective Bayesian prior distribution given two simple postulates is the exponential of the entropy of the probability distribution (e^S). These postulates are based on well accepted properties of probabilities [10]. The proof involves the uniqueness of differential equation solutions [11], mathematical induction [12], and the ϵ-δ definition of the convergence of a function [13].

This objective prior distribution represents the statistical weight that any given probability distribution will occur naturally. It is given by the mathematical symbol ρ which is consistent with ρ's meaning in physics as the density of states [14]. ρ will, in general, be an improper prior distribution, meaning that it is not normalized initially. Since ρ takes in a probability function and returns a number in the range zero to infinity, it is a functional.

The first postulate that is required to derive ρ is that the prior distribution is symmetric in its indices. Intuitively, this states that the statistical weight of a bag with an 80% chance of drawing a blue marble and a 20% chance of drawing a red marble must be the same as the statistical weight of a bag with a 20% change of drawing a blue marble and an 80% chance of drawing a red marble. This is because the ordering of the probabilities is chosen by the statistician and does not effect the underlying properties of the data. The first postulate as an equation is,

$$\rho(\cdots, p_j, \cdots, p_k, \cdots) = \rho(\cdots, p_k, \cdots, p_j, \cdots) \tag{1}$$

The second postulate that is required to derive ρ is that the prior distribution of two statistically independent probabilities is the multiple of the two independent prior distributions. For two statistically separate distributions, the joint distribution is the combination of them through multiplication [10]. Likewise, the statistical weight that independent events can occur together is the multiple of the statistical weight that they can occur independently. For example, say that there is a bag with blue and red marbles in it and a completely separate bag with green and yellow marbles in it. Each of these bags represents a probability distributions based on the ratio of colored marbles inside of them. If there are n ways to produce a probability distribution with P_{blue} and P_{red} and m ways to produce the probability distribution P_{green} and P_{yellow}, then for each of the n distributions there are m distributions. This means that the joint distribution which describes the probability of drawing red and green, red and yellow, blue and green, or blue and yellow has n times m ways to exist. The second postulate as an equation is,

$$\rho(P, Q) = \rho(P) \times \rho(Q) \tag{2}$$

Each point on the probability distribution is independent from the rest [10]. Due to the second postulate (2), ρ across the entire distribution is given by the multiplication of the statistical weight that a given probability can happen at each point.

$$\rho(P) = \Pi_i \sigma_i(p_i) \tag{3}$$

Where $\sigma_i(p_i)$ is the prior distribution of the single state's probability p_i. By the first postulate equation (1), the location of p_j and p_k can be switched without changing $\rho(P)$. Switching p_j and p_k in (3) along with canceling out identical terms on each side yields,

$$\frac{\sigma_j(p_k)}{\sigma_k(p_k)} = \frac{\sigma_j(p_j)}{\sigma_k(p_j)} \tag{4}$$

Since the left hand side of the equation is not a function of p_j, both sides must evaluate to a constant K_0.

$$K_0 = \frac{\sigma_j(p_j)}{\sigma_k(p_j)} \rightarrow \sigma_j(p_j) = K_0 \sigma_k(p_j) \tag{5}$$

This means that whenever the ordering of two σ_i terms are switched there comes a factor of K_0. Switching the order of σ_j and σ_k twice (one switch forward and one switch back) returns the initial ρ but with an extra factor of K_0^2.

$$K_0^2 \times \rho(P) = \rho(P) \rightarrow K_0^2 = 1 \tag{6}$$

Solving this yields,

$$K_0 = \pm 1 \tag{7}$$

K_0 must be greater than or equal to zero for ρ to represent the statistical weight of a distribution because ρ must always be a number greater than or equal to zero. This gives that K_0 is positive, so $K_0 = 1$. Plugging this result into (5) yields,

$$\sigma_j(p_j) = \sigma_k(p_j) \tag{8}$$

Thus every σ_i is the same function at every location in the probability distribution,

$$\rho(P) = \Pi_i^n \sigma(p_i) \tag{9}$$

Since σ represents the statistical weight that a given probability distribution will exist in nature, it is a number in the range $[0, \infty)$. This allows the definition of a new function, γ, that is in the range $(-\infty, \infty)$ such that,

$$\sigma = e^{\gamma} \tag{10}$$

This definition of γ will help simplify the math later on. Equation (10) can be plugged into (9) to get,

$$\rho(P) = \Pi_i \sigma(p_i) = e^{\sum_i \gamma(p_i)} \tag{11}$$

A new function, $\omega(P)$, can be defined by taking the natural log of (11) as so,

$$\omega(P) = \ln(\rho) = \ln\big(e^{\sum_i \gamma(p_i)}\big) = \sum_i \gamma(p_i) \tag{12}$$

Evaluating this at a joint distribution between two statistically independent distributions from postulate (2) gives the formula,

$$\omega(P,Q) = \omega(P) + \omega(Q) \rightarrow \sum_i \sum_j \gamma(p_i q_j) = \big(\sum_i \gamma(p_i)\big) + \big(\sum_j \gamma(q_j)\big) \tag{13}$$

The unique solution to equation 13 will now be derived. Solving for ω using any one given probability distribution will give ω for them all since the σs are all the same by (9) and (12). For simplicity, take the special case that both P and Q are the following probability distributions,

$$\begin{aligned} P &= (p, 1-p) \\ Q &= (q, 1-q) \end{aligned} \tag{14}$$

with both p and q arbitrary numbers in the range $[0,1]$. This allows (13) to be written as,

$$\gamma\big(pq\big) + \gamma\big(p(1-q)\big) + \gamma\big(q(1-p)\big) + \gamma\big((1-p)(1-q)\big) = \gamma\big(p\big) + \gamma\big(1-p\big) + \gamma\big(q\big) + \gamma\big(1-q\big) \tag{15}$$

The derivative with respect to p yields,

$$q\gamma'\big(pq\big) + (1-q)\gamma'\big(p(1-q)\big) - q\gamma'\big(q(1-p)\big) - (1-q)\gamma'\big((1-p)(1-q)\big) = \gamma'\big(p\big) - \gamma'\big(1-p\big) \tag{16}$$

The derivative with respect to q yields,

$$\begin{aligned} \gamma'\big(pq\big) + pq\gamma''\big(pq\big) - \gamma'\big(p(1-q)\big) - p(1-q)\gamma''\big(p(1-q)\big) - \gamma'\big(q(1-p)\big) \\ -q(1-p)\gamma''\big(q(1-p)\big) + \gamma'\big((1-p)(1-q)\big) + (1-p)(1-q)\gamma''\big((1-p)(1-q)\big) = 0 \end{aligned} \tag{17}$$

The only solution to this equation is at,

$$x\gamma''(x) + \gamma'(x) + f(x) = 0 \tag{18}$$

With,

$$f(pq) - f(q(1-p)) - f(p(1-q)) + f((1-p)(1-q)) = 0 \tag{19}$$

The partial derivative of (19) with respect to p yields,

$$qf'(pq) + qf'(q(1-p)) - (1-q)f'(p(1-q)) - (1-q)f'((1-p)(1-q)) = 0 \tag{20}$$

The partial derivative with respect to q yields,

$$\begin{aligned} pqf''(pq) + f'(pq) + q(1-p)f''(q(1-p)) + f'(q(1-p)) + f'(p(1-q)) + p(1-q)f''(p(1-q)) \\ +(1-q)(1-q)f''((1-p)(1-q)) + f'((1-p)(1-q)) = 0 \end{aligned} \tag{21}$$

The only solution to this equation is,

$$xf''(x) + f'(x) + g(x) = 0 \tag{22}$$

With the constraint,

$$\sum g(P,Q) = g(pq) + g(q(1-p)) + g(p(1-q)) + g((1-p)(1-q)) = 0 \tag{23}$$

In order to solve (22) the function g must be known.

2.1. Lemma that g is Zero

It will now be proven that the only valid solution for $g(P)$ from (22) is the constant function that outputs zero everywhere. Using the constraint equation (23) on a probability distribution with the two probabilities p and $1-p$ gives,

$$g(p) + g(1-p) = 0 \rightarrow g(p) = -g(1-p) \tag{24}$$

Evaluating (24) at $p = \frac{1}{2}$ yields,

$$g(\frac{1}{2}) = -g(1-\frac{1}{2}) = -g(\frac{1}{2}) \rightarrow g(\frac{1}{2}) = 0 \tag{25}$$

Evaluating the constraint equation (23) at the probability distribution $\frac{1}{4}, \frac{1}{4}, \frac{1}{2}$ gives,

$$\begin{aligned} g(\frac{1}{4}) + g(\frac{1}{4}) + g(\frac{1}{2}) = 0 \\ \rightarrow 2g(\frac{1}{4}) + 0 = 0 \rightarrow g(\frac{1}{4}) = 0 \end{aligned} \tag{26}$$

By repeating this process the following equation can be derived,

$$g\big((\frac{1}{2})^n\big) = 0 \tag{27}$$

For $n \geq 0$ and an integer. Equation (27) will be shown to be true using proof by mathematical induction. The case $n = 0$ yields,

$$g\big(1\big) = 0 \tag{28}$$

Which is true since it is just (23) evaluated at the probability distribution with one possible outcome that has 100% chance to occur. As is common in induction, it will now be assumed that (27) is true for $n-1$ and all smaller non-negative integers and it will be shown that (27) must then be also true for n. The probability distribution $(\frac{1}{2})^n, (\frac{1}{2})^n, (\frac{1}{2})^{n-1}, \cdots (\frac{1}{2})$ is a valid probability distribution since it sums to one. This can be seen by inspecting the geometric sum [15],

$$\begin{aligned} \sum_{i=1}^{i=n} (\frac{1}{2})^i = 1 - (\frac{1}{2})^n \rightarrow (\frac{1}{2})^n + \sum_{i=1}^{i=n} (\frac{1}{2})^i \\ = (\frac{1}{2})^n + (\frac{1}{2})^n + (\frac{1}{2})^{n-1} + \cdots (\frac{1}{2}) = 1 \end{aligned} \tag{29}$$

Evaluating the constraint equation (23) at the probability distribution $(\frac{1}{2})^n, (\frac{1}{2})^n, (\frac{1}{2})^{n-1}, \cdots (\frac{1}{2})$ yields,

$$g\big((\frac{1}{2})^n\big) + \sum_{i=1}^{i=n} g\big((\frac{1}{2})^i\big) = 0 \tag{30}$$

All of the terms except the $g\big((\frac{1}{2})^n\big)$ terms are zero by the induction assumption,

$$g\big((\frac{1}{2})^n\big) + g\big((\frac{1}{2})^n\big) + \sum_{i=1}^{i=n} 0 = 0 \rightarrow g\big((\frac{1}{2})^n\big) = 0 \tag{31}$$

This concludes the proof by induction.

A corollary must be show which will be used later to help prove that any arbitrary point of g is zero. It will be proven that any real number in the range $[0,1]$ can be expressed as the sum of some subset of the set of numbers $(\frac{1}{2})^i$ for i a positive integer,

$$\alpha = \sum_i \gamma_i (\frac{1}{2})^i \tag{32}$$

With γ_i being 0 or 1 at any given i and α an arbitrary real number in the range $[0,1]$. To prove this it will be shown that for a sufficiently large number of terms added into the sum the distance between α and the sum can be made arbitrarily small. This is Cauchy's definition of ϵ-δ convergence [13].

Define the current distance from the sum up to j terms to α as,

$$d(j) = \alpha - \sum_{i=1}^{i=j} \gamma_i (\frac{1}{2})^i \tag{33}$$

It is asserted that,

$$d(j) \leq (\frac{1}{2})^j \tag{34}$$

This means that d can be made arbitrarily small for large enough j. Equation (34) will be proven with mathematical induction [12]. First take the base case that $j = 0$,

$$d(0) \leq (\frac{1}{2})^0 = 1 \rightarrow d(0) \leq 1 \tag{35}$$

The sum has no terms in it and therefore the sum is zero. This means,

$$d(0) = \alpha \tag{36}$$

α is in the range $[0, 1]$ and any number in this range is less than or equal to one, so (35) must be true. To continue the proof by induction, it will now be assumed that (34) is true for $j = n - 1$. The goal is to show that it must also be true for n. By the induction assumption, the distance at $(n - 1)$ must obey the inequality,

$$d(n-1) \leq (\frac{1}{2})^{n-1} \tag{37}$$

at $j = n$ either add in $(\frac{1}{2})^n$ or zero to the sum that had previously ended at $j = n - 1$. If

$$d(n-1) \leq (\frac{1}{2})^n \tag{38}$$

Then set $\gamma_n = 0$ and thus do not add in a new term into the sum. If this is true then the proof is concluded. Otherwise,

$$d(n-1) \geq (\frac{1}{2})^n \tag{39}$$

In this case add in the $(\frac{1}{2})^n$ term by setting $\gamma_n = 1$. Using (33) yields,

$$d(n) = d(n-1) - (\frac{1}{2})^n \tag{40}$$

By the induction assumption (37) it is true that,

$$d(n) \leq (\frac{1}{2})^{n-1} - (\frac{1}{2})^n = (\frac{1}{2})^n \rightarrow d(n) \leq (\frac{1}{2})^n \tag{41}$$

This is what was set out to be shown, thus concluding the induction proof. This means that $d(n)$ can be made arbitrarily small for a large enough j and thus the sum can be made to converge to any real number in the range $[0, 1]$ with the appropriate choice of the γ_is and sufficiently large j.

It follows that the number $1 - q \in [0, 1]$ can be written in terms of a sum,

$$1 - q = \sum_i \gamma_i (\frac{1}{2})^i \tag{42}$$

Applying the constraint equation (23) at the probability distribution q and $1 - q$ along with (42) yields,

$$\begin{aligned} 0 &= g(q) + \sum_i \gamma_i \times g\big((\frac{1}{2})^i\big) \\ &= g(q) + \sum_i \gamma_i \times 0 \rightarrow g(q) = 0 \end{aligned} \tag{43}$$

Where (27) was used to zero out the sum $\sum_i \gamma_i \times g\big((\frac{1}{2})^i\big)$. Since q was arbitrary and it was just shown that $g(q) = 0$, any and all arbitrary points on g must equal zero. Thus the proof that g must always be zero is concluded.

2.2. *Continued Derivation of Objective Bayesian Prior Distribution*

Equation (43) can be plugged into (22) to get,

$$xf''(x) + f'(x) + 0 = 0 \tag{44}$$

This can be solved using standard differential equation techniques [16],

$$f(p) = c_1 \ln(p) + c_2 \tag{45}$$

Equation (45) can be plugged into (18) to yield,

$$x\gamma''(x) + \gamma'(x) + c_1 \ln(x) + c_2 = 0 \tag{46}$$

This differential equation has the solution [16],

$$\gamma(p) = c_3 + 2c_1 p - c_1 p \ln(p) - c_2 p + c_3 \ln(p) \tag{47}$$

This solution can be evaluated at (13),

$$\sum_{i=1}^{i=n} \sum_{j=1}^{j=m} \big(c_3 + 2c_1 P_i Q_j - c_1 P_i Q_j \ln(P_i Q_j) - c_2 P_i Q_j + c_3 \ln(P_i Q_j)\big) = \tag{48}$$

$$\big(\sum_{i=1}^{i=n} c_3 + 2c_1 P_i - c_1 P_i \ln(P_i) - c_2 P_i + c_3 \ln(P_i)\big) + \big(\sum_{j=1}^{j=m} c_3 + 2c_1 Q_j - c_1 Q_j \ln(Q_j) - c_2 Q_j + c_3 \ln(Q_j)\big)$$

Simplifying yields,

$$2c_1 + mnc_3 - c_2 + mc_3 \sum_{i=1}^{i=n} \ln(P_i) + nc_3 \sum_{j=1}^{j=m} \ln(Q_j) - c_1 \sum_{i=1}^{i=n} P_i \ln(P_i) - c_1 \sum_{j=1}^{j=m} Q_j \ln(Q_j) = \tag{49}$$

$$nc_3 + 2c_1 - c_2 - c_1 \sum_{i=1}^{i=n} P_i \ln(P_i) + c_3 \sum_{i=1}^{i=n} \ln(P_i) + mc_3 + 2c_1 - c_2 - c_1 \sum_{j=1}^{j=m} Q_j \ln(Q_j) + c_3 \sum_{j=1}^{j=m} \ln(Q_j)$$

Canceling out terms yields,

$$-2c_1 + c_2 + c_3\big(mn - n - m + (m-1) \sum_{i=1}^{i=n} \ln(P_i) + (n-1) \sum_{j=1}^{j=m} \ln(Q_j)\big) = 0 \tag{50}$$

The only way that (50) can be true for an arbitrary probability distribution is if $2c_1 = c_2$ and $c_3 = 0$. Plugging this into (47) gives the solution for γ,

$$\gamma(p) = -c_1 p \ln(p) \tag{51}$$

Equation (51) means that γ is an arbitrary constant times the entropy [6], measured in Shannon's, of the probability distribution. Plugging this result into (11) yields,

$$\rho(P) = e^{-c_1 \sum p \ln(p)} = e^{c_1 S_{\text{Shannon's}}} = \Pi_i p_i^{-c_1 p_i} \tag{52}$$

Due to the uniqueness of differential equation solutions [11], this is the unique solution for the objective prior probability distribution, $\rho(P)$, given the two postulates (1) and (2).

2.3. Relationship with Information Theory

Entropy is the average amount of information in a probability distribution [6]. Given n bits of information it is possible to encode 2^n different states [17]. An example of this is that there are, $256 = 2^8$, 8-bit numbers. Equation (52) can be interpreted as, it is possible to encode,

$$\rho(P) = 2^{c_1 S_{\text{Bits}}} \tag{53}$$

different probability distributions with S bits of information. Where ρ is the number of states of the information. Setting $c_1 = 1$ allows a direct relationship between the number of ways that a probability distribution can exist and the information inside of it.

$$\rho(P) = 2^{S_{\text{bits}}} = e^{S_{\text{Shannons}}} = \Pi_i p_i^{-p_i} \tag{54}$$

Where S_{bits} is the entropy measured in bits and S_{Shannons} is the entropy measured in Shannons. Any other value of c_1 will not make the units of the entropy line up with the base of the exponent. Equation (54) is asserted to be the correct expression for the objective Bayesian prior distribution given the two postulates (1) and (2) and is therefore the main result of this paper.

Probability distributions with larger entropy have a higher statistical weight due to (54). This means that the most likely probability distributions are the maximal entropy ones. Thus, (54) confirms the maximal entropy principle used in statistics and thermodynamics [14, 18]. In the next section, it will be shown how to use (54) to boost the accuracy of modern day artificial intelligence by adding in a novel regularization term to the loss function of neural networks [7, 9].

3. Application of Novel Objective Prior to Neural Networks

A common problem in data science is to fit sample points to a probability distribution that can then be used to make inferences about future samples. One way to do this is to use a neural network [9], which is a randomly initialized guess of a function that is refined using gradient decent [8]. A common loss functions to fit a neural network model to a data set is cross entropy [19]. In this section, the explicit formula for the ideal Bayesian estimator that minimizes the cross entropy loss function is derived. This will then be used in conjunction with (54) to describe a new technique to regularize neural networks that use cross entropy as a loss function.

3.1. General Formula For Bayesian Estimator

The average error between a chosen probability distribution and all distributions weighted by both the likelihood that they produces the sample data points and the probability that the distribution exists is [1],

$$\mathcal{E}(\{\vec{x}\}_s, \rho, Q) = \int_P \mathcal{L}(P, \{\vec{x}\}_s) \times \rho(P) \times H(P, Q)[dP] \tag{55}$$

Where $\int_P [dP]$ stands for the functional integral over all valid probability distributions (functions whose sum is equal to one and values are real and non-negative), $\mathcal{L}(P, \{\vec{x}\}_s)$ is the likelihood that a probability distributions (P) gives the set of sample data ($\{\vec{x}\}_s$), $\rho(P)$ is the prior probability that a given probability distribution will occur, and $H(P, Q)$ is the

loss function between P and Q. Picking the appropriate Q to minimize this Bayesian loss gives the Bayesian estimator, which is the mathematically ideal probability distribution given this sample data, prior probability, and loss function.

3.2. Minimizing Cross Entropy

The cross entropy between the probability distributions P and Q will be minimized. Cross entropy loss is the amount of information required to encoding the drawing of a sample point from P assuming that a probability distribution Q generated the data [19]. The explicit formula for the loss is,

$$H(P,Q) = -\sum_i P_i \times \ln\big(Q_i\big) \tag{56}$$

Equation (55) for the average loss will be used along with Lagrangian multipliers [20] to extremize the cross entropy loss over all probability functions P subject to the constraint that the statistician chosen probability, Q, must sum to one. The equation to extremize is,

$$\mathcal{E}(\{\vec{x}\}_s, Q, \lambda_1) = \int_P \mathcal{L}(P,\{\vec{x}\}_s) \times \rho(P) \times H(P,Q)[dP] + \lambda_1\big(\sum_{\vec{x}} Q(\vec{x}) - 1\big) \tag{57}$$

Where λ_1 is the Lagrangian multiplier. Extremizing the equation leads to the solution,

$$Q(\vec{x}',\rho) = \frac{\int_P \mathcal{L}(P,\{\vec{x}\}_s) \times \rho(P) \times P(\vec{x}')[dP]}{\int_P \mathcal{L}(P,\{\vec{x}\}_s) \times \rho(P)[dP]} \tag{58}$$

Where this is true for all possible values of $\vec{x}'$. The denominator is a normalizing factor that can be found at the end by imposing normalization. This $Q(\vec{x}',\rho)$ is the ideal Bayesian estimator for cross entropy loss, the set of sample data $\{\vec{x}\}_s$ and prior distribution ρ. Equation (58) can be seen as the average of all probability distributions weighted by both how often each individual distribution occurs and how often it produces the sample data.

The P that is dominate in the functional integral (58) is the probability distribution that is most likely to have produced the data. This is the maximum of the following functional,

$$\mathcal{L}(P,\{\vec{x}\}_s) \times \rho(P) \tag{59}$$

Taking the natural log, dividing by the number of data samples in $\{\vec{x}\}_s$, and multiplying by negative one to turn this functional into a loss that must be minimized.

$$\begin{aligned} L(P,\{\vec{x}\}_s,\rho) &= -\frac{1}{N}\ln\big(\mathcal{L}(P,\{\vec{x}\}_s) \times \rho(P)\big) \\ &= H(\{\vec{x}\}_s, P) - \frac{1}{N}\ln\big(\rho(P)\big) \end{aligned} \tag{60}$$

Where $H(\{\vec{x}\}_s, P)$ is the familiar cross entropy loss, and N is the number of sample points. This is the standard loss used in many categorical neural networks but with the addition of one extra term, $-\frac{1}{N}\ln\big(\rho(P)\big)$. To stay true to Bayesian statistics, a $-\frac{1}{N}\ln\big(\rho(P)\big)$ must be included into the loss function as a regularization term. This will lead to more accurate prediction of the model on unseen data [7]. Using the solution for ρ from (54) yields,

$$L(P,\{\vec{x}\}_s) = H(\{\vec{x}\}_s, P) - \frac{1}{N}S(P) \tag{61}$$

With S being the entropy of the probability distribution measured in Shannon's. This novel extra regularization term ($-\frac{1}{N}S(P)$) in the loss function will be called, "Entropy Regularization".

3.3. Result of Entropy Regularization on Real World Data

In this subsection, Entropy Regularization from (61) will be used on real world data. The nature of Entropy Regularization makes it force the model to be less confident that a given sample is in a class. This means that it may behave similarly to other known regularization techniques such as lowering the weight of samples in a training set that performed well and raising the weight of under performing samples. The specifics of its performance are given below.

The results of Entropy Regularization from (61) may vary in effectiveness because: it does not include additional information from prepossessing, the metric that is used to evaluate the performance of a model may not be cross entropy, gradient decent algorithms are only an approximation of full Bayesian statistics, and Entropy Regularization should be effective on average over many different data sets but may not perform perfectly on any one individual data set. This is why the limits of its effectiveness are on real world data [21–30] should be tested using neural networks [9].

Three data sets where chosen to test Entropy Regularization on: the MNIST Handwritten Digits data set [21], the Credit Card Fraud data set [22–29] and the Graduate Admission's data set [30]. Each data set has categorical targets. The average f1 score [31] across all classes was used to rank the models. In the first nine runs, standard regularization techniques of AiZiA's Integral Regularization [32], Dropout Regularization [33], and L2 Kernel Regularization [34] were used. On the first runs, a few different parameters for each regularization technique were tried. In the next run Entropy Regularization was used alone. In the last three runs, Entropy Regularization was used along with the other three techniques utilizing the parameters that gave the best results from the first few runs. The highest f1 score on held out validation data is made bold in the tables below to show which regularization technique performed best on the given data. The same neural network architecture, preprossessing, optimizer [35], and data

augmentation were used on each data set to keep the results of every run comparable.

3.3.1. MNIST Handwritten Digits

The MNIST handwritten digits data set [21] is comprised of 42,000 handwritten digits from zero through nine. Each sample is a 28 by 28, grey scale, 8-bit, image of a number from zero through nine. The data was prepossessed by rescaling it into the range of minus three to three. Each image was augmented every epoch with random: resizing, cropping, brightness, and contrast changes. The neural network architecture had initial convolution layers that fed into dense layers. There are five convolution layers, a maxpooling layer, and eight dense layers. Cross entropy loss was used to classify images into ten categories, one for each number.

Table 1. *Results of various different neural network models trained on the MNIST Handwritten Digits data set. The numbers in the columns represent the average f1 score of the model for a given regularization type and value of its regularization parameters for a given run. The best result is boldfaced.*

Index of Run	Entropy Used	None	Integral	Dropout	Kernel
1	False	0.9912	0.9949	0.9900	0.9944
2	False	N/A	0.9930	0.9921	0.9945
3	False	N/A	0.9951	0.9907	0.9934
4	True	0.9898	0.9940	0.9923	0.9940

Integral Regularization [32] out performed the rest with Kernel Regularization [34] as a close second. Entropy Regularization from (61) helped Dropout [33] perform better, but still did not do as well as the other techniques.

3.3.2. Credit Card Fruad Detection

The Credit Card Fraud data set [22–29] is taken from real world European credit card fraud cases. The data set contains 284,807 credit card transactions, 30 features per sample, and highly imbalanced classes. The data was standardized and then fed into a fully connected dense thirty seven layered neural network architecture. Cross entropy loss was used to classify transactions as: either "fraud" or "not fraud".

Table 2. *Results of various different neural network models trained on the Credit Card Fraud data set. The numbers in the columns represent the average f1 score of the model for a given regularization type and value of its regularization parameters for a given run. The best result is boldfaced.*

Index of Run	Entropy Used	None	Integral	Dropout	Kernel
1	False	0.5650	0.6091	0.7995	0.5653
2	False	N/A	0.8883	0.8725	0.4996
3	False	N/A	0.00173	0.8458	0.4996
4	True	0.7055	0.6649	0.4996	0.7308

Integral Regularization [32] out performed the rest of the techniques with Dropout [33] coming in a close second. Entropy Regularization from (61) improved the results of the model when either no other regularization was used or when Kernel Regularization [34] was used.

3.3.3. Graduate Admissions

The U.S. Graduate Admissions data set [30] is comprised of 500 sample points of real world students data with seven features per sample. The features include GPA, test scores, and research experience. The data was prepossessed with principal component analysis and then fed into a dense neural network for predictions about the chance of admission or rejection for a student. The network has six dense layers. Cross entropy loss was used to classify samples as either, "will be admitted to graduate school" or, "will be rejected from graduate school".

Table 3. *Results of various different neural network models trained on the Graduate Admissions data set. The numbers in the columns represent the average f1 score of the model for a given regularization type and value of its regularization parameters for a given run. The best result is boldfaced.*

Index of Run	Entropy Used	None	Integral	Dropout	Kernel
1	False	0.8957	1	1	1
2	False	N/A	1	0.8198	0.9241
3	False	N/A	1	0.7818	0.8957
4	True	1	1	0.8957	1

All four regularization techniques received a perfect f1 score of one.

3.3.4. Summary of Real World Data Set Findings

Entropy Regularization from (61) interacted well with the other regularization techniques. Often including it into the neural network training boosted the accuracy of the model. It did lower the f1 score on a few trials though, so it should be used with care in general. As is always the case, it is important to not overly regularize the neural network or the training loss may rise above acceptable levels. This means that if Entropy Regularization is included into a model, then it may be best to lower the amount of other regularization that was used previously.

4. Conclusion

In this paper, a novel objective prior distribution was derived to be the exponential of the entropy information in a probability distribution (e^S) [6]. This allowed it to be related to information theory, the maximal entropy principal, and thermodynamics [14, 18]. This novel objective prior distribution was then used to derive a new regularization technique [7] that could be used in artificial intelligence gradient decent algorithms [8] based on updating the loss function with a term that is the negative of the entropy divided by the number of data points [6]. The regularization technique theoretically works best when there is very little information about how the sample data was produced. Including Entropy Regularization into several neural network models sometimes boosted their f1 score [31] on three real world test data [21–30]. In a few trials the f1 score was lowered due to over regularization, so some level of care must be used before applying it to real world artificial intelligence models. This technique was derived assuming categorical data, so it should not be used on regression data sets.

Future work includes expanding the understanding of when and how to use this new objective prior distribution from (54) in real world applications including, but not limited to,

data science, thermodynamics, and optimization algorithms. The limits of Entropy Regularization were not fully explored here, so more testing of it may lead to additional valuable insights. There is a also a connection between the Feynman Path Integral of quantum mechanics [36] and thermodynamics. Since this paper connects statistics and thermodynamics, it may be possible to form a connection from statistics and information theory directly to quantum mechanics.

Acknowledgements

Mary G. Watson, PhD, retired, thanks for your statistical insights and encouragement. John and Betsy Watson thanks for your unending support.

References

[1] BAYES, "AN ESSAY TOWARDS SOLVING A PROBLEM IN THE DOCTRINE OF CHANCES," *Biometrika*, vol. 45, pp. 296–315, 12 1958.

[2] H. Jeffreys, "An invariant form for the prior probability in estimation problems," *Royal Society*, vol. 186, 1946.

[3] J. B. S. Haldane, "A note on inverse probability," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 28, no. 1, pp. 55–61, 1932.

[4] J. M. Bernardo, "Reference posterior distributions for bayesian inference," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 41, no. 2, pp. 113–128, 1979.

[5] M. Ghosh, "Objective Priors: An Introduction for Frequentists," *Statistical Science*, vol. 26, no. 2, pp. 187 – 202, 2011.

[6] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[7] P. Buhlmann and S. Van De Geer, *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.

[8] C. Lemaréchal, "Cauchy and the gradient method," *Doc Math Extra*, vol. 251, no. 254, p. 10, 2012.

[9] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

[10] S. Ross, *A First Course in Probability*. Pearson Prentice Hall, 2010.

[11] D. Zill, *A First Course in Differential Equations (5th ed.)*. 2001.

[12] A. BOUHOULA, E. KOUNALIS, and M. RUSINOWITCH, "Automated Mathematical Induction," *Journal of Logic and Computation*, vol. 5, pp. 631–668, 10 1995.

[13] W. Felscher, "Bolzano, cauchy, epsilon, delta," *The American Mathematical Monthly*, vol. 107, no. 9, pp. 844–862, 2000.

[14] F. Mandl, *Statistical Physics*. Manchester Physics Series, Wiley, 2013.

[15] J. Babb, "Mathematical concepts and proofs from nicole oresme: Using the history of calculus to teach mathematics," *Science and Education*, vol. 14, pp. 443–456, 07 2005.

[16] W. R. Inc., "Differential equation solver." Champaign, IL, 2021.

[17] A. Ralston and E. D. Reilly, *Encyclopedia of Computer Science (3rd Ed.)*. USA: Van Nostrand Reinhold Co., 1993.

[18] E. T. Jaynes, "Information theory and statistical mechanics," *Physical review*, vol. 106, no. 4, p. 620, 1957.

[19] K. P. Murphy, *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press, 2013.

[20] D. Luenberger, *Linear and Nonlinear Programming: Second Edition*. Springer US, 2003.

[21] "Digit recognizer." https://www.kaggle.com/c/digit-recognizer/overview. Accessed: 2021-06-18.

[22] "Credit card fraud detection." https://www.kaggle.com/mlg-ulb/creditcardfraud. Accessed: 2021-06-18.

[23] A. Dal Pozzolo, O. Caelen, R. Johnson, and G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," 12 2015.

[24] A. Dal Pozzolo, O. Caelen, Y.-A. Le Borgne, S. Waterschoot, and G. Bontempi, "Learned lessons in credit card fraud detection from a practitioner perspective," *Expert Systems with Applications*, vol. 41, pp. 4915–4928, 08 2014.

[25] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, pp. 1–14, 09 2017.

[26] F. Carcillo, A. Dal Pozzolo, Y.-A. Le Borgne, O. Caelen, Y. Mazzer, and G. Bontempi, "Scarff : a scalable framework for streaming credit card fraud detection with spark," *Information Fusion*, vol. 41, 09 2017.

[27] B. Lebichot, Y.-A. Le Borgne, L. He, F. Oble, and G. Bontempi, *Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection*, pp. 78–88. 01 2019.

[28] F. Carcillo, Y.-A. Le Borgne, O. Caelen, Y. Kessaci, F. Oble, and G. Bontempi, "Combining unsupervised and supervised learning in credit card fraud detection," *Information Sciences*, 05 2019.

[29] Y.-A. Le Borgne and G. Bontempi, *Machine Learning for Credit Card Fraud Detection - Practical Handbook*. 05 2021.

[30] "Us graduate school's admission parameters." https://www.kaggle.com/tanmoyie/us-graduate-schools-admission-parameters. Accessed: 2021-06-18.

[31] D. Powers, "Evaluation: From precision, recall and f-factor to roc, informedness, markedness and correlation," *Mach. Learn. Technol.*, vol. 2, 01 2008.

[32] "Aizia." https://www.aizia.org/. Accessed: 2021-06-18.

[33] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012.

[34] S. BÃ$\frac{1}{4}$hlmann, Peter; van de Geer, *Statistics for High-Dimensional Data [electronic resource] : Methods, Theory and Applications*. 2011.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[36] R. P. Feynman, "Space-time approach to non-relativistic quantum mechanics," *Feynman's Thesis A New Approach To Quantum Theory*, pp. 71–109, 2005.