# Why Softmax? Because It Is the Only Consistent Approach to Probability-Based Classification*

Anatole Lokshin[1], Olga Kosheleva[2][0000−0003−2587−4209], and Vladik Kreinovich[2][0000−0002−1244−1650]

[1] Alpine Replay
214 7th Street, Huntington Beach, CA 92648, USA
anatole@traceup.com
[2] University of Texas at El Paso, El Paso, TX 79968, USA
{olgak,vladik}@utep.edu

**Abstract.** In many practical problems, the most effective classification techniques are based on deep learning. In this approach, once the neural network generates values corresponding to different classes, these values are transformed into probabilities by using the softmax formula. Researchers tried other transformation, but they did not work as well as softmax. A natural question is: why is softmax so effective? In this paper, we provide a possible explanation for this effectiveness: namely, we prove that softmax is the only consistent approach to probability-based classification. In precise terms, it is the only approach for which two reasonable probability-based ideas – Least Squares and Bayesian statistics – always lead to the exact same classification.

**Keywords:** Classification · Machine learning · Softmax · Least Squares · Bayesian approach.

## 1 Formulation of the Problem

**Classification problems are ubiquitous.** In many practical situations, we need to classify an image or the situation into one of the given categories. For example:

- a security system needs to decide whether an incoming email is legitimate or malicious,
- a medical imaging system needs to decide whether a tumor is benign or malignant,

* This work was supported in part by the National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), HRD-1834620 and HRD-2034030 (CAHSI Includes), EAR-2225395, and by the AT&T Fellowship in Information Technology.
It was also supported by the program of the development of the Scientific-Educational Mathematical Center of Volga Federal District No. 075-02-2020-1478, and by a grant from the Hungarian National Research, Development and Innovation Office (NRDI).

– am environmental imaging system needs to decide what animal it sees, etc.

In many cases, we cannot identify the class with a 100% accuracy. In such cases, we would like to know the probabilities $p_1, \ldots, p_N$ of different classes – probabilities that add to 1: $p_1 + \ldots + p_n = 1$.

Usually, the reason why we want to classify is that we need to make a decision based on this classification:

– whether to pass the email to the user,
– whether to perform a surgery, etc.

A natural idea is to make a decision corresponding to the most probable class, i.e., to the class with the largest probability $p_i$.

*Comment.* Probability values corresponding to other classes indicate how confident we should be in this decision.

**How these problems are solved now: enter deep learning.** At present, for many problems, the most effective methods are methods based on deep learning; see, e.g., [2]. Classification problems are no exception: in many cases, deep learning leads to the most successful classification.

**We need to transform values – generated by deep neural networks – into probabilities.** Traditionally, deep learning techniques are used for machine learning:

– We have several examples $(x_1^{(k)}, \ldots, x_m^{(k)}, y^{(k)})$ $(k = 1, \ldots, K)$ in which we know both the inputs $x_1^{(k)}, \ldots, x_m^{(k)}$ and the desired output $y^{(k)}$.
– We have a new case, in which we only know the input $x_1, \ldots, x_m$.
– Based on this information, we want to predict the output $y$ corresponding to this input.

The predicted value $\widetilde{y}$ is what the deep neural network generates – after being trained on the $K$ known examples.

In the classification problem, for each of the $K$ training examples, we know whether the corresponding object belongs to this class or not. For each class, we thus have the output $y^{(k)}$ to be:

– either 1 – if the object belongs to this class,
– of 0 – if the object does not belong to this class.

For each of $n$ classes, we can train the corresponding neural network. Thus, for each new object described by the values $x_1, \ldots, x_m$, and for each class $i$, we get some value $y_i$ that, crudely speaking, describes the computer's confidence that the corresponding object belongs to theis class. The larger this value, the more confident is the computer systems.

These values can be negative – e.g., slightly smaller than 0, these values can be larger than 1. And in almost all cases, they do not add up to 1. But what we need are probabilities – i.e., we need $n$ non-negative numbers that add up

to 1. Thus, we need to transform the values $v_i$ generated from deep learning into probabilities.

**How values are transformed into probabilities: general idea.** First, we need to make sure that the values are non-negative. For this purpose, we select a measurable (e.g., continuous) function $s(y)$ that transforms the whole real line into the set of non-negative numbers. By using this function, we transform the original $n$ real values $y_1, \ldots, y_n$ into $n$ non-negative numbers $s(y_1), \ldots, s(y_n)$.

Then, to come up with non-negative values that add up to 1, we divide each of the resulting numbers by their sum:

$$p_i = \frac{s(y_i)}{\sum\limits_{j=1}^{n} s(y_j)}.$$

*Comment.* For this formula to be always applicable, we need to make sure that all the values $s(y)$ are positive. Indeed, if we have $s(y) = 0$ for some $y$, and we encounter a situation in which $y_1 = \ldots = y_n = y$, then the above formula leads to the undefined value $0/0$.

**How values are transformed into probabilities: specifics.** In principle, we can use different functions $s(y)$. It turns out that in classification problems, the most effective function is the exponential function $\exp(k \cdot y)$, for an appropriate value $k > 0$. The resulting transformation into probabilities

$$p_i = \frac{\exp(k \cdot y_i)}{\sum\limits_{j=1}^{n} \exp(k \cdot y_j)}$$

is known as *softmax*. This name comes from the fact that:

- instead of simply crisply selecting the class that provides the largest degree of confidence $y_i$,
- we make a softer choice, and allow other options with some probability,
- although, of course, the probability of these options is smaller than the probability of the largest-confidence class.

*Comment.* Instead of $s(y) = \exp(k \cdot y)$, one can use a function $s(y) = C \cdot \exp(k \cdot y)$ for some $C > 0$; then, we will get the exact same probabilities $p_i$: indeed, if we plug in the new function into the general expression for $p_i$, the factors $C$ in the numerator and in the denominator of this expression cancel out, and we get the exact same expression for $p_i$ as without the factor $C$.

**Remaining problem and what we do about it.** Why the exponential function works better than other possible monotonic function is a mystery. In this paper, we provide a possible explanation for this mysterious fact: namely, we show that the exponential function corresponding to softmax is the only one that leads to a consistent probability-based classification.

## 2 Our Explanation

**Dynamic classification: a problem.** To explain our idea, we need to consider frequent situations in which we need to classify an object based in a dynamical situation, when:

- instead of a single image,
- we have several images corresponding to different moments of time.

For example, in an environmental system, we have several blurry images of an animal taken at several consequent moments of time. Based on this information, we need to identify the animal.

Let $T$ denote the number of consequent images. Let us number them in chronological order by numbers $t = 1, \ldots, T$. To each of these images $t$, we can apply the classification algorithm and come up with the probability values $p_{1,t}, \ldots, p_{n,t}$. Based on all these values, we need to come up with the probabilities $p_1, \ldots, p_n$. How can we combine $T$ tuples of probability values into a single tuple?

**Two approaches to solving this problem: general idea.** We will show that standard probability ideas lead to two natural approaches to solving this problem.

- We will prove that for softwax, when we use an exponential function $s(y)$, these two approaches lead to the exact same selection of the most probable class. In this sense, softmax leads to a consistent assignment of probabilities.
- We will also prove that for any function $s(y)$ that is different from softmax-based exponential function, these two approaches sometimes lead to different selections. In this sense, softmax is the *only* approach that leads to a consistent assignment of probabilities.

Let us describe these two approaches.

**First approach: using Least Squares (LS) approach.** For each class $i$ and for each moment $t$, the neural network generates a value $y_{i,t}$. This value describes the system's degree of confidence – based on the observation at moment $t$ – that the observed object belongs to the class $i$. A natural interpretation is that there is some actual (unknown) degree of confidence $y_i$, and all $T$ values $y_{i,1}, \ldots, y_{i,T}$ are estimates of this true value.

In these terms, the problem of estimating $y_i$ becomes a particular case of the general problem of estimating the value $q$ of a quantity based on several observations $q_1, \ldots, q_T$. In other words, we have $T$ approximate equalities $q \approx q_1$, $\ldots$, $q \approx q_T$. In this general situation, a natural idea is to use the Least Squares approach, i.e., to find the estimate $q$ that minimizes the sum

$$(q - q_1)^2 + \ldots + (q - q_T)^2;$$

see, e.g., [3]. To find the optimal value $q$, we can differentiate the minimized expression by $q$ and equate the derivative to 0. This leads to the arithmetic

average

$$q = \frac{q_1 + \ldots + q_T}{T}.$$

In our case, we get

$$y_i = \frac{y_{i,1} + \ldots + y_{i,T}}{T}. \tag{1}$$

Based on these values, we can compute the corresponding probabilities

$$p_i^{\mathrm{LS}} = \frac{s(y_i)}{\sum\limits_{j=1}^{n} s(y_j)}.$$

*Comment.* In statistics, the Least Squares method is usually justified by assuming that the approximation errors are independent and normally distributed. This is a reasonable assumption, but it is not always satisfied.

However, there is a more general common-sense justification of the Least Squares approach. Namely, $T$ approximate equalities $q \approx q_1$, $\ldots$, $q \approx q_T$ can be summarized by saying that the tuple $(q, \ldots, q)$ should be close to the tuple $(q_1, \ldots, q_T)$. In this formulation, it is natural to select the value $q$ for which the distance between these two tuples is the smallest – or, equivalently, for which the square of this distance is the smallest. And this square of the distance is exactly the Least Squares sum

$$(q - q_1)^2 + \ldots + (q - q_T)^2.$$

**Second approach: using Bayesian (B) approach.** Another idea is using the Bayesian approach; see, e.g., [3]. Here, we have $n$ hypotheses corresponding to $n$ classes. In the general case, a priori, we do not know which class is more probable. This means that each class must be assigned the same prior probability $p_0(i) = 1/n$. According to Bayes formula, once we have observations $E$, the probability changes to

$$p_i^{\mathrm{B}} = \frac{p(E \,|\, i) \cdot p_0(i)}{\sum\limits_{j=1}^{n} p(E \,|\, j) \cdot p_0(j)},$$

where $p(E \,|\, i)$ is the conditional probability that we observe $E$ under the condition that the actual class is $i$.

Since all prior probabilities are the same $p_0(1) = \ldots = p_0(n) = 1/n$, we can divide both the numerator and the denominator by this common probability, and get a simplified formula

$$p_i^{\mathrm{B}} = \frac{p(E \,|\, i)}{\sum\limits_{j=1}^{n} p(E \,|\, j)},$$

For each image $t$, the probability that we observe this image under this condition is equal to $p_{i,t}$. Similarly to the least squares approach, it is reasonable to assume that the observations are independent – in the sense that the corresponding approximation errors are independent. Under this assumption, the probability that we observe all $T$ images under the hypothesis $i$ is equal to the product of individual probabilities, i.e., we have $p(E \,|\, i) = p_{i,1} \cdot \ldots \cdot p_{i,T}$. In this case, the above formula for $p_i$ takes the form

$$p_i^{\mathrm{B}} = \frac{p_{i,1} \cdot \ldots \cdot p_{i,T}}{\sum\limits_{j=1}^{n} p_{j,1} \cdot \ldots \cdot p_{j,T}}. \tag{2}$$

**Main result.** Now, we are ready to formulate our main result.

**Definition 1.** *By a* transformation*, we mean a measurable continuous function whose values are all positive.*

**Definition 2.**

- *By* data*, we mean a triple $D = \langle n, T, \{y_{i,t}\}_{i=1,\ldots,n;t=1,\ldots,T}\rangle$, where $n$ and $T$ are positive integers and $y_{i,t}$ are real numbers.*
- *Integers from 1 to $n$ will be called* classes*.*

**Definition 3.** *Let $s(y)$ be a transformation, and let $D$ be data.*

- *We say that the class $i_0$ is* LS-most probable *if $p_{i_0}^{\mathrm{LS}} = \max_i p_i^{\mathrm{LS}}$, where*

$$p_i^{\mathrm{LS}} \stackrel{\mathrm{def}}{=} \frac{s(y_i)}{\sum\limits_{j=1}^{n} s(y_j)}$$

  *and the value $y_i$ is determined by the formula* (1)*.*
- *We say that the class $i_0$ is* B-most probable *if $p_{i_0}^{\mathrm{B}} = \max_i p_i^{\mathrm{B}}$, where $p_i^{\mathrm{B}}$ is determined by the formula* (2)*, in which*

$$p_{i,t} \stackrel{\mathrm{def}}{=} \frac{s(y_{i,t})}{\sum\limits_{j=1}^{n} s(y_{j,t})}.$$

**Proposition.** *For each transformation $s(y)$, the following two conditions are equivalent to each other:*

- *For each data $D$, a class $i_0$ is LS-most probable if and only if it is B-most probable.*
- *The transformation $s(y)$ is equal to $C \cdot \exp(k \cdot y)$ for some constants $C$ and $k$.*

*Comment.* Thus indeed, the exponential function – corresponding to softmax – is the only one for which the two probabilistic approaches lead to the same result. This explains why this function is so successful in practice – it is the only function that leads to a consistent probability-based classification.

**Proof.**

1°. Let us first prove that if we use the exponential function as the transformation, then a class $i_0$ is LS-most probable if and only if it is B-most probable.

Indeed, all the expressions $p_i$ have the same denominator, so the inequality $p_{i_0}^{\mathrm{LS}} \geq p_i^{\mathrm{LS}}$ describing LS-comparison is equivalent to $s(y_{i_0}) \geq s(y_i)$. Since the function $s(y)$ is strictly increasing, this inequality, in its turn, is equivalent to $y_{i_0} \geq y_i$. By definition of the values $y_i$, this means that

$$\frac{y_{i_0,1} + \ldots + y_{i_0,T}}{T} \geq \frac{y_{i,1} + \ldots + y_{i,T}}{T}.$$

If we multiply both sides of this inequality by $T$, we get an equivalent inequality

$$y_{i_0,1} + \ldots + y_{i_0,T} \geq y_{i,1} + \ldots + y_{i,T}.$$

The function $\exp(k \cdot y)$ with $k > 0$ is strictly increasing, so this inequality is equivalent to

$$\exp(k \cdot (y_{i_0,1} + \ldots + y_{i_0,T})) \geq \exp(k \cdot (y_{i,1} + \ldots + y_{i,T})).$$

Here,

$$\exp(k \cdot (y_{i,1} + \ldots + y_{i,T})) = \exp(k \cdot y_{i,1}) \cdot \ldots \cdot \exp(k \cdot y_{i,T}) = s(y_{i,1}) \cdot \ldots \cdot s(y_{i,T}),$$

so we have

$$s(y_{i_0,1}) \cdot \ldots \cdot s(y_{i_0,T}) \geq s(y_{i,1}) \cdot \ldots \cdot s(y_{i,T}).$$

Dividing both sides by the product of the same denominators $\sum\limits_{j=1}^{n} s(y_{j,t})$ corresponding to $t = 1, \ldots, T$, we get an equivalent inequality

$$p_{i_0,1} \cdot \ldots \cdot p_{i_0,T} \geq p_{i,1} \cdot \ldots p_{i,T}.$$

Finally, dividing both sides by the same sum $\sum_{j=1}^{n} p_{j,1} \cdot \ldots \cdot p_{j,T}$, we get the equivalent inequality $p_{i_0}^{\mathrm{B}} \geq p_i^{\mathrm{B}}$. So, for the exponential transformation, indeed, a class is LS-optimal if and only if it is B-optimal.

2°. Let us now prove that if we for all data, a class $i_0$ is LS-most probable if and only if it is B-most probable, then the transformation is the exponential function.

Indeed, let $s(y)$ be a transformation for which a class $i_0$ is LS-most probable if and only if it is B-most probable. For every two real numbers $a$ and $b$, let us consider the data in which $n = T = 2$, and

$$y_{1,1} = 1, \quad y_{1,2} = b, \quad y_{2,1} = a + b, \quad y_{2,2} = 0.$$

For this data,

$$y_1 = y_2 = \frac{a+b}{2},$$

so

$$p_1^{\mathrm{LS}} = p_2^{\mathrm{LS}} = \frac{s(y_i)}{s(y_1)+s(y_2)} = \frac{1}{2}.$$

So, in this case, both classes $i = 1$ and $i = 2$ are LS-most probable. Thus, because of our assumption about $s(z)$, they must both be B-most probable. By definition of what is B-most probable, this means that we must have $p_1^{\mathrm{B}} = p_2^{\mathrm{B}}$. By the formula (2), this means that

$$\frac{p_{1,1}\cdot p_{1,2}}{p_{1,1}\cdot p_{1,2}+p_{2,1}\cdot p_{2,2}} = \frac{p_{2,1}\cdot p_{2,2}}{p_{1,1}\cdot p_{1,2}+p_{2,1}\cdot p_{2,2}}.$$

Multiplying both sides by the common denominator, we conclude that

$$p_{1,1}\cdot p_{1,2} = p_{2,1}\cdot p_{2,2}. \tag{3}$$

Here,

$$p_{1,1} = \frac{s(a)}{s(a)+s(a+b)}, \quad p_{1,2} = \frac{s(b)}{s(b)+s(0)},$$
$$p_{2,1} = \frac{s(a+b)}{s(a)+s(a+b)}, \quad p_{2,2} = \frac{s(0)}{s(b)+s(0)}.$$

So, the equality (3) takes the form

$$\frac{s(a)\cdot s(b)}{(s(a)+s(a+b))\cdot(s(b)+s(0))} = \frac{s(a+b)\cdot s(0)}{(s(a)+s(a+b))\cdot(s(b)+s(0))}.$$

Multiplying both sides by the common denominator, we get

$$s(a)\cdot s(b) = s(a+b)\cdot s(0).$$

All the values of the function $s(y)$ are positive. So, we can apply logarithm to both sides of this equality, and get

$$L(a)+L(b) = L(a+b)+L(0),$$

where we denoted $L(y) \stackrel{\mathrm{def}}{=} \ln(s(y))$, so that $s(y) = \exp(L(y))$.

If we subtract $2L(0)$ from both sides, we get

$$L(a)-L(0)+L(b)-L(0) = L(a+b)-L(0),$$

i.e.,

$$F(a)+F(b) = F(a+b), \tag{4}$$

where we denoted $F(y) \stackrel{\mathrm{def}}{=} L(y)-L(0)$ (so that $L(y) = F(y)+L(0)$).

By definition, the transformation $s(y)$ is measurable. Thus, the functions $L(y) = \ln(s(y))$ and $F(y) = L(y)-L(0)$ are also measurable. It is known (see,

e.g., [1]) that if a measurable function satisfies the property (4) for all $a$ and $b$, then it is a linear function, i.e., $F(y) = k \cdot y$ for some $k$. Thus, $L(y) = F(y) + L(0) = L(0) + k \cdot y$, and

$$s(y) = \exp(L(y)) = \exp(L(0) + k \cdot y) = C \cdot \exp(k \cdot y),$$

where we denoted $C \stackrel{\text{def}}{=} \exp(L(0))$.

The proposition is proven.

## References

1. J. Aczél and J. Dhombres, *Functional Equations in Several Variables*, Cambridge University Press, 2008.
2. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016.
3. D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.