The Fundamental Nature of the Log Loss Function

Vladimir Vovk
Department of Computer Science, Royal Holloway, University of London, Egham, Surrey, UK

June 15, 2015

Abstract

The standard loss functions used in the literature on probabilistic prediction are the log loss function, the Brier loss function, and the spherical loss function; however, any computable proper loss function can be used for comparison of prediction algorithms. This note shows that the log loss function is most selective in that any prediction algorithm that is optimal for a given data sequence (in the sense of the algorithmic theory of randomness) under the log loss function will be optimal under any computable proper mixable loss function; on the other hand, there is a data sequence and a prediction algorithm that is optimal for that sequence under either of the two other loss functions but not under the log loss function.

Keywords: algorithmic theory of randomness, mixability, predictive complexity, predictive randomness, probabilistic prediction, proper loss functions.

1 Introduction

In his work Yuri Gurevich has emphasized practical aspects of algorithmic randomness. In particular, he called for creating a formal framework allowing us to judge whether observed events can be regarded as random or point to something dubious going on (see, e.g., the discussion of the lottery organizer's wife winning the main prize in [5]). The beautiful classical theory of randomness started by Andrey Kolmogorov and Per Martin-Löf has to be restricted in order to achieve this goal and avoid its inherent incomputabilities and asymptotics.

This note tackles another practically-motivated question: what are the best loss functions for evaluating probabilistic prediction algorithms? Answering this question, however, requires extending rather than restricting the classical theory of algorithmic randomness.

In the empirical work on probabilistic prediction in machine learning (see, e.g., [2]) the most standard loss functions are log loss and Brier loss, and spherical loss is a viable alternative; all these loss functions will be defined later in this note. It is important to understand which of these three loss functions is likely to lead to better prediction algorithms. We formalize this question using a generalization of the notion of Kolmogorov complexity called predictive complexity (see, e.g., [7]; it is defined in Section 3). Our answer is that the log loss function is likely to lead to better prediction algorithms as it is more selective: if a prediction algorithm is optimal under the log loss function, it will be optimal under the Brier and spherical loss functions, but the opposite implications are not true in general.

As we discuss at the end of Section 3, the log loss function corresponds to the classical theory of randomness. Therefore, our findings confirm once again the importance of the classical theory and are not surprising at all from the point of view of that theory. But from the point of view of experimental machine learning, our recommendation to use the log loss function rather than Brier or spherical is less trivial.

This note is, of course, not the first to argue that the log loss function is fundamental. For example, David Dowe has argued for it since at least 2008 ([3], footnote 175; see [4], Section 4.1, for further references). Another paper supporting the use of the log loss function is Bickel's [1].

2 Loss Functions

We are interested in the problem of binary probabilistic prediction: the task is to predict a binary label $y \in \{0,1\}$ with a number $p \in [0,1]$; intuitively, p is the predicted probability that $y = 1$. The quality of the prediction p is measured by a *loss function* $\lambda : [0,1] \times \{0,1\} \to \mathbb{R} \cup \{+\infty\}$. Intuitively, $\lambda(p,y)$ is the loss suffered by a prediction algorithm that outputs a prediction p while the actual label is y; the value $+\infty$ (from now on abbreviated to ∞) is allowed. Following [10], we will write $\lambda_y(p)$ in place of $\lambda(p,y)$, and so identify λ with the pair of functions (λ_0, λ_1) where $\lambda_0 : [0,1] \to \mathbb{R} \cup \{+\infty\}$ and $\lambda_1 : [0,1] \to \mathbb{R} \cup \{+\infty\}$. We will assume that $\lambda_0(0) = \lambda_1(1) = 0$, that the function λ_0 is increasing, that the function λ_1 is decreasing, and that $\lambda_y(p) < \infty$ unless $p \in \{0,1\}$.

A loss function λ is called *η-mixable* for $\eta \in (0,\infty)$ if the set

$$\left\{(u,v) \in [0,1]^2 \mid \exists p \in [0,1] : u \le e^{-\eta\lambda(p,0)} \text{ and } v \le e^{-\eta\lambda(p,1)}\right\}$$

is convex; we say that λ is *mixable* if it is η-mixable for some η.

A loss function λ is called *proper* if, for all $p, q \in [0,1]$,

$$\mathbb{E}_p\, \lambda(p,\cdot) \le \mathbb{E}_p\, \lambda(q,\cdot), \tag{1}$$

where $\mathbb{E}_p\, f := pf(1) + (1-p)f(0)$ for $f : \{0,1\} \to \mathbb{R}$. It is *strictly proper* if the inequality in (1) is strict whenever $q \ne p$.

We will be only interested in computable loss functions (the notion of computability is not defined formally in this note; see, e.g., [7]). We will refer to the loss functions satisfying the properties listed above as *CPM* (computable proper mixable) loss functions.

Besides, we will sometimes make the following *smoothness assumptions*:

- λ_0 is infinitely differentiable over the interval $[0,1)$ (the derivatives at 0 being one-sided);
- λ_1 is infinitely differentiable over the interval $(0,1]$ (the derivatives at 1 being one-sided);
- for all $p \in (0,1)$, $(\lambda_0'(p), \lambda_1'(p)) \neq 0$.

We will refer to the loss functions satisfying all the properties listed above as *CPMS* (computable proper mixable smooth) loss functions.

Examples

The most popular loss functions in machine learning are the *log loss function*

$$\lambda_1(p) := -\ln p, \quad \lambda_0(p) := -\ln(1-p)$$

and the *Brier loss function*

$$\lambda(p,y) := (y-p)^2.$$

Somewhat less popular is the *spherical loss function*

$$\lambda_1(p) := 1 - \frac{p}{\sqrt{p^2 + (1-p)^2}}, \quad \lambda_0(p) := 1 - \frac{1-p}{\sqrt{p^2 + (1-p)^2}}.$$

All three loss functions are mixable, as we will see later. They are also computable (obviously), strictly proper (this can be checked by differentiation), and satisfy the smoothness conditions (obviously). Being computable and strictly proper, these loss functions can be used to measure the quality of probabilistic predictions.

Mixability and Propriety

Intuitively, propriety can be regarded as a way of parameterizing loss functions, and we get it almost for free for mixable loss functions. The essence of a loss function is its *prediction set*

$$\{(\lambda_0(p), \lambda_1(p)) \mid p \in [0,1]\}. \tag{2}$$

When given a prediction set, we can parameterize it by defining $(\lambda_0(p), \lambda_1(p))$ to be the point (x,y) of the prediction set at which $\inf_{(x,y)}(py + (1-p)x)$ is attained. This will give us a proper loss function. And if the original loss function satisfies the smoothness conditions (and so, intuitively, the prediction set does not have corners), the new loss function will be strictly proper.

3 Repetitive Predictions

Starting from this section we consider the situation, typical in machine learning, where we repeatedly observe data $z_1, z_2, \ldots$ and each observation $z_t = (x_t, y_t) \in \mathbf{Z} = \mathbf{X} \times \{0,1\}$ consists of an *object* $x_t \in \mathbf{X}$ and its *label* $y_t \in \{0,1\}$. Let us assume, for simplicity, that $\mathbf{X}$ is a finite set, say a set of natural numbers.

A *prediction algorithm* is a computable function $F : \mathbf{Z}^* \times \mathbf{X} \to [0,1]$; intuitively, given a data sequence $\sigma = (z_1, \ldots, z_T)$ and a new object x, F outputs a prediction $F(\sigma, x)$ for the label of x. For any data sequence $\sigma = (z_1, \ldots, z_T)$ and loss function λ, we define the cumulative loss that F suffers on σ as

$$\mathrm{Loss}_F^\lambda(\sigma) := \sum_{t=1}^{T} \lambda(F(z_1, \ldots, z_{t-1}, x_t), y_t)$$

(where $z_t = (x_t, y_t)$ and $\infty + a$ is defined to be ∞ for any $a \in \mathbb{R} \cup \{\infty\}$). Functions $\mathrm{Loss}_F^\lambda : \mathbf{Z}^* \to \mathbb{R}$ that can be defined this way for a given λ are called *loss processes under* λ. In other words, $L : \mathbf{Z}^* \to \mathbb{R}$ is a loss process under λ if and only if $L(\Box) = 0$ (where $\Box$ is the empty sequence) and

$$\forall \sigma \in \mathbf{Z}^* \, \forall x \in \mathbf{X} \, \exists p \in [0,1] \, \forall y \in \{0,1\} : L(\sigma, x, y) = L(\sigma) + \lambda(p, y). \tag{3}$$

A function $L : \mathbf{Z}^* \to \mathbb{R}$ is said to be a *superloss process under* λ if (3) holds with $\geq$ in place of $=$. If λ is computable and mixable, there exists a smallest, to within an additive constant, upper semicomputable superloss process:

$$\exists L_1 \, \forall L_2 \, \exists c \in \mathbb{R} \, \forall \sigma \in \mathbf{Z}^* : L_1(\sigma) \leq L_2(\sigma) + c,$$

where L_1 and L_2 range over upper semicomputable superloss processes under λ. (For a precise statement and proof, see [7], Theorem 1, Lemma 6, and Corollary 3; [7] only considers the case of a trivial one-element $\mathbf{X}$, but the extension to the case of general $\mathbf{X}$ is easy.) For each computable mixable λ (including the log, Brier, and spherical loss functions), fix such a smallest upper semicomputable superloss process; it will be denoted $\mathcal{K}^\lambda$, and $\mathcal{K}^\lambda(\sigma)$ will be called the *predictive complexity* of $\sigma \in \mathbf{Z}^*$ under λ. The intuition behind $\mathcal{K}^\lambda(\sigma)$ is that this is the loss of the ideal prediction strategy whose computation is allowed to take an infinite amount of time.

In this note we consider infinite data sequences $\zeta \in \mathbf{Z}^\infty$, which are idealizations of long finite data sequences. If $\zeta = (z_1, z_2, \ldots) \in \mathbf{Z}^\infty$ and T is a nonnegative integer, we let ζ^T to stand for the prefix $z_1 \ldots z_T$ of ζ of length T.

The *randomness deficiency* of $\sigma \in \mathbf{Z}^*$ with respect to a prediction algorithm F under a computable mixable loss function λ is defined to be

$$D_F^\lambda(\sigma) := \mathrm{Loss}_F^\lambda(\sigma) - \mathcal{K}^\lambda(\sigma); \tag{4}$$

since Loss_F^λ is upper semicomputable ([7], Section 3.1), the function $D_F^\lambda : \mathbf{Z}^* \to \mathbb{R}$ is bounded below. Notice that the indeterminacy $\infty - \infty$ never arises in (4) as $\mathcal{K}^\lambda < \infty$. We will sometimes replace the upper index λ in any of the three terms of (4) by "ln" in the case where λ is the log loss function.

Let us say that $\zeta \in \mathbf{Z}^\infty$ is *random with respect to F under λ* if

$$\sup_T D_F^\lambda(\zeta^T) < \infty.$$

The intuition is that in this case F is an optimal prediction algorithm for ζ under λ.

Log Randomness

In the case where λ is the log loss function and $\mathbf{X}$ is a one-element set, the predictive complexity of a finite data sequence σ (which is now a binary sequence if we ignore the uninformative objects) is equal, to within an additive constant, to $-\ln M(\sigma)$, where M is Levin's *a priori* semimeasure. (In terms of this note, a *semimeasure* can be defined as a process of the form e^{-L} for some superloss process L under the log loss function; *Levin's* a priori *semimeasure* is a largest, to within a constant factor, lower semicomputable semimeasure.) The randomness deficiency $D_F^{\ln}(\sigma)$ of σ with respect to a prediction algorithm F is then, to within an additive constant, $\ln(M(\sigma)/P(\sigma))$, where P is the probability measure corresponding to F,

$$P(y_1, \dots, y_T) := \bar{p}_1 \cdots \bar{p}_T, \quad \bar{p}_t := \begin{cases} F(y_1, \dots, y_{t-1}) & \text{if } y_t = 1 \\ 1 - F(y_1, \dots, y_{t-1}) & \text{if } y_t = 0 \end{cases}$$

(we continue to ignore the objects, which are not informative). Therefore, $D_F^{\ln}(\sigma)$ is a version of the classical randomness deficiency of σ, and $\zeta \in \{0,1\}^\infty$ is random with respect to F under the log loss function if and only if ζ is random with respect to P in the sense of Martin-Löf.

4 A Simple Statement of Fundamentality

In this section, we consider computable proper mixable loss functions.

Theorem 1. *Let λ be a CPM loss function. If a data sequence $\zeta \in \mathbf{Z}^\infty$ is random under the log loss function with respect to a prediction algorithm F, it is random under λ with respect to F.*

A special case of this theorem is stated as Proposition 16 in [13].

Let us say that a CPM loss function λ is *fundamental* if it can be used in place of the log loss function in Theorem 1. The proof of the theorem will in fact demonstrate its following quantitative form: for any computable $\eta > 0$ and any computable proper η-mixable λ there exists a constant c_λ such that, for any prediction algorithm F,

$$D_F^{\ln} \ge \eta D_F^\lambda - c_\lambda. \tag{5}$$

Let us define the *mixability constant* η_λ of a loss function λ as the supremum of η such that λ is η-mixable. It is known that a mixable loss function λ is

η_λ-mixable ([12], Lemmas 10 and 12); therefore, (5) holds for $\eta = \eta_\lambda$, provided η_λ is computable.

If $\mathbf{X}$ is a one-element set (and so the objects do not play any role and can be ignored), the notion of randomness under the log loss function coincides with the standard Martin-Löf randomness, as discussed in the previous section. Theorem 1 shows that other notions of randomness are either equivalent or weaker.

A *superprediction* is a point in the plane that lies Northeast of the prediction set (2) (i.e., a point $(x, y) \in \mathbb{R}^2$ such that $\lambda_0(p) \le x$ and $\lambda_1(p) \le y$ for some $p \in [0, 1]$).

of Theorem 1. We will prove (5) for a fixed $\eta \in (0, \infty)$ such that η is computable and λ is η-mixable. Let L be a superloss process under λ and F be a prediction algorithm. Fix temporarily $(\sigma, x) \in \mathbf{Z}^* \times \mathbf{X}$ and set $p := F(\sigma, x) \in [0, 1]$; notice that $(a, b) := (L(\sigma, x, 0) - L(\sigma), L(\sigma, x, 1) - L(\sigma))$ is a λ-superprediction. By the definition of η-mixability there exists a parallel translation of the curve $e^{-\eta x} + e^{-\eta y} = 1$ that passes through the point $\lambda^p := (\lambda_0(p), \lambda_1(p))$ and lies Southeast of the prediction set of λ. Let h be the affine transformation of the plane mapping that translation onto the curve $e^{-x} + e^{-y} = 1$; notice that h is the composition of the scaling $(x, y) \mapsto \eta(x, y)$ by η and then parallel translation moving the point $\eta\lambda^p$ to the point $(-\ln(1-p), -\ln p)$. The λ-superprediction (a, b) is mapped by h to the ln-superprediction

$$(\eta a + (-\ln(1-p)) - \eta\lambda_0(p), \eta b + (-\ln p) - \eta\lambda_1(p)) .$$

We can see that $\eta L + \mathrm{Loss}_F^{\ln} - \eta \mathrm{Loss}_F^{\lambda}$ is a superloss process under ln. It is clear that this ln-superloss process is upper semicomputable if L is. Therefore, for some constant c_λ,

$$\mathcal{K}^{\ln} \le \eta\mathcal{K}^{\lambda} + \mathrm{Loss}_F^{\ln} - \eta \mathrm{Loss}_F^{\lambda} + c_\lambda,$$

which is equivalent to (5). □

5 A Criterion of Fundamentality

In this section, we only consider computable proper mixable loss functions that satisfy, additionally, the smoothness conditions. The main result of this section is the following elaboration of Theorem 1 for CPMS loss functions.

Theorem 2. *A CPMS loss function λ is fundamental if and only if*

$$\inf_p (1-p)\lambda_0'(p) > 0. \tag{6}$$

Equivalently, it is fundamental if and only if

$$\inf_p (-p)\lambda_1'(p) > 0. \tag{7}$$

We can classify CPMS loss functions λ by their *degree*

$$\deg(\lambda) := \inf\left\{k : \lambda_0^{(k)}(0) \neq 0 \text{ and } \lambda_1^{(k)}(1) \neq 0\right\},$$

where $^{(k)}$ stands for the kth derivative and, as usual, $\inf \emptyset := \infty$. We will see later in this section that Theorem 2 can be restated to say that the fundamental loss functions are exactly those of degree 1. Furthermore, we will see that for a CPMS loss function λ of degree $1 < k < \infty$ there exist a data sequence $\zeta \in \mathbf{Z}^\infty$ and a prediction algorithm F such that ζ is random with respect to F under λ while the randomness deficiency $D_F^{\ln}(\zeta^T)$ of ζ^T with respect to F under the log loss function grows almost as fast as $T^{1-1/k}$ as $T \to \infty$.

Straightforward calculations show that the log loss function has degree 1 and the Brier and spherical loss functions have degree 2.

In the proof of Theorem 2 we will need the notion of the *signed curvature* of the *prediction curve* $(\lambda_0(p), \lambda_1(p))$ at a point $p \in (0,1)$, which can be defined as

$$k_\lambda(p) := \frac{\lambda_0'(p)\lambda_1''(p) - \lambda_1'(p)\lambda_0''(p)}{(\lambda_0'(p)^2 + \lambda_1'(p)^2)^{3/2}}. \tag{8}$$

The mixability constant η_λ (i.e., the largest η for which λ is η-mixable) is

$$\eta_\lambda = \inf_p \frac{k_\lambda(p)}{k_{\ln}(p)}.$$

Therefore, λ is mixable if and only if

$$\inf_p \frac{k_\lambda(p)}{k_{\ln}(p)} > 0. \tag{9}$$

Lemma 1. *A CPMS loss function λ is fundamental if and only if*

$$\sup_p \frac{k_\lambda(p)}{k_{\ln}(p)} < \infty$$

(cf. (9)).

The proof the part "if" of Lemma 1 goes along the same lines as the proof of Theorem 1, and also shows that, if λ and Λ are CPMS loss functions such that

$$\eta_\lambda := \inf_p \frac{k_\lambda(p)}{k_{\ln}(p)} > 0 \quad \text{and} \quad H_\Lambda := \sup_p \frac{k_\Lambda(p)}{k_{\ln}(p)} < \infty$$

are computable numbers, then there exists $c_{\lambda,\Lambda} \in \mathbb{R}$ such that, for any prediction algorithm F,

$$H_\Lambda D_F^\Lambda \geq \eta_\lambda D_F^\lambda - c_{\lambda,\Lambda}.$$

We will call H_Λ the *fundamentality constant* of Λ (analogously to η_λ being called the mixability constant of λ).

Notice that the log loss function (perhaps scaled by multiplying by a positive constant) is the only loss function for which the mixability and fundamentality

constants coincide, $\eta_{\ln} = H_{\ln}$. Therefore, fundamental CPMS loss functions can be regarded as log-loss-like.

The part "only if" of Lemma 1 will be proved below, in the proof of Theorem 2.

The computation of k_λ for the three basic loss functions using (8) gives:

- For the log loss function, the result is

$$k_{\ln}(p) = \frac{p(1-p)}{(p^2 + (1-p)^2)^{3/2}}. \tag{10}$$

- For the Brier loss function, the result is

$$k_{\text{Brier}}(p) = \frac{1}{2}\frac{1}{(p^2 + (1-p)^2)^{3/2}}.$$

- For the spherical loss function, the result is

$$k_{\text{spher}}(p) = 1.$$

We can plug the expression (10) for the signed curvature of the log loss function into Lemma 1 to obtain a more explicit statement. Because of the propriety of λ, this statement can be simplified, which gives the following corollary.

Corollary 1. *A CPMS loss function λ is fundamental if and only if*

$$\sup_p \frac{\lambda_0'(p)\lambda_1''(p) - \lambda_1'(p)\lambda_0''(p)}{\lambda_0'(p)\lambda_1'(p)(\lambda_1'(p) - \lambda_0'(p))} < \infty. \tag{11}$$

Proof. In view of the expressions (8) and (10), the condition in Lemma 1 can be written as

$$\sup_p \frac{\lambda_0'(p)\lambda_1''(p) - \lambda_1'(p)\lambda_0''(p)}{(\lambda_0'(p)^2 + \lambda_1'(p)^2)^{3/2}} \frac{(p^2 + (1-p)^2)^{3/2}}{p(1-p)} < \infty.$$

Therefore, it suffices to check that

$$\frac{(\lambda_0'(p)^2 + \lambda_1'(p)^2)^{3/2}}{\lambda_0'(p)\lambda_1'(p)(\lambda_1'(p) - \lambda_0'(p))} = \frac{(p^2 + (1-p)^2)^{3/2}}{p(1-p)}.$$

The last equality follows from

$$\frac{\lambda_1'(p)}{\lambda_0'(p)} = \frac{p-1}{p}, \tag{12}$$

which in turn follows from the propriety of λ. □

It is instructive to compare the criterion (11) with the well-known criterion

$$\inf_p \frac{\lambda_0'(p)\lambda_1''(p) - \lambda_1'(p)\lambda_0''(p)}{\lambda_0'(p)\lambda_1'(p)(\lambda_1'(p) - \lambda_0'(p))} > 0 \tag{13}$$

for λ being mixable (see, e.g., [6] or [8], Theorem 2; it goes back to [11], Lemma 1). The criterion (13) can be derived from (9) as in the proof of Corollary 1.

of Theorem 2. Differentiating (12) we obtain

$$\frac{\lambda_1''(p)\lambda_0'(p) - \lambda_1'(p)\lambda_0''(p)}{\lambda_0'(p)^2} = p^{-2},$$

and the fundamentality constant (11) of λ is

$$\begin{aligned}\sup_p \frac{p^{-2}\lambda_0'(p)^2}{\lambda_0'(p)\lambda_1'(p)(\lambda_1'(p) - \lambda_0'(p))} &= \sup_p \frac{p^{-2}}{\lambda_0'(p)(\lambda_1'(p)/\lambda_0'(p))(\lambda_1'(p)/\lambda_0'(p) - 1)} \\ &= \sup_p \frac{p^{-2}}{\lambda_0'(p)(1 - 1/p)(-1/p)} = \sup_p \frac{1}{\lambda_0'(p)(1-p)},\end{aligned}$$

where we have used (12). This gives us (6); in combination with (12) we get (7).

Let us now prove the part "only if" of Theorem 2 (partly following the argument given after Proposition 16 of [13]). According to (12), (6) and (7) are equivalent. Suppose that

$$\inf_p (1-p)\lambda_0'(p) = 0,$$

and let us check that λ is not fundamental. By the smoothness assumptions, we have $(1-p)\lambda_0'(p) = 0$ either for $p = 0$ or for $p = 1$. Suppose, for concreteness, that $(1-p)\lambda_0'(p) = 0$ for $p = 0$ (if $(1-p)\lambda_0'(p) = 0$ for $p = 1$, we will have $(-p)\lambda_1'(p) = 0$ for $p = 1$, and we can apply the same argument as below for $p = 1$ in place of $p = 0$). Let k be such that $\lambda_0^{(k)}(0) > 0$ but $\lambda_0^{(i)}(0) = 0$ for all $i < k$; we know that $k \geq 2$ (the easy case where $\lambda_0^{(i)}(0) = 0$ for all i should be considered separately). Consider any data sequence $\zeta = (x_1, y_1, x_2, y_2, \ldots) \in \mathbf{Z}^\infty$ in which all labels are 0: $y_1 = y_2 = \cdots = 0$. We then have $\sup_T \mathcal{K}^{\ln}(\zeta^T) < \infty$ and $\sup_T \mathcal{K}^{\lambda}(\zeta^T) < \infty$. Let F be the prediction algorithm that outputs $p_t := t^{-1/k-\epsilon}$ at step t, where $\epsilon \in (0, 1 - 1/k)$. Then ζ is random with respect to F under λ since the loss of this prediction algorithm over the first T steps is

$$\sum_{t=1}^{T} \lambda_0(p_t) \leq 2 \sum_{t=1}^{T} \frac{\lambda_0^{(k)}(0)}{k!} p_t^k + O(1)$$

(we have used Taylor's approximation for λ_0) and the series $\sum_t p_t^k$ is convergent. On the other hand, the randomness deficiency of ζ^T with respect to F under the log loss function grows as

$$-\sum_{t=1}^{T} \ln(1 - p_t) \sim \sum_{t=1}^{T} p_t \sim \frac{k}{k - 1 - k\epsilon} T^{1-1/k-\epsilon}. \qquad \square$$

□

Notice that the criterion of mixability (13) can be simplified when we use (12): it becomes

$$\sup_p (1-p)\lambda_0'(p) < \infty$$

or, equivalently,

$$\sup_p (-p)\lambda_1'(p) < \infty.$$

The function $(1-p)\lambda_0'(p) = (-p)\lambda_1'(p)$ can be computed as

- 1 in the case of the log loss function;
- $2p(1-p)$ in the case of the Brier loss function;
- $p(1-p)(p^2+(1-p)^2)^{-3/2}$ in the case of the spherical loss function.

Therefore, all three loss functions are mixable, but only the log loss function is fundamental.

It is common in experimental machine learning to truncate allowed probabilistic predictions to the interval $[\epsilon, 1-\epsilon]$ for a small constant $\epsilon > 0$ (this boils down to cutting off the ends of the prediction sets corresponding to the slopes below ϵ and above $1-\epsilon$). It is easy to check that in this case all CPMS loss functions lead to the same notion of randomness.

Corollary 2. *CPMS loss functions λ and Λ restricted to $p \in [\epsilon, 1-\epsilon]$, where $\epsilon > 0$, lead to the same notion of randomness.*

We can make the corollary more precise as follows: for prediction algorithms F restricted to $[\epsilon, 1-\epsilon]$, D_F^λ and D_F^Λ coincide to within a factor of

$$\max\left(\sup_{p\in[\epsilon,1-\epsilon]} \frac{k_\lambda(p)}{k_\Lambda(p)}, \sup_{p\in[\epsilon,1-\epsilon]} \frac{k_\Lambda(p)}{k_\lambda(p)}\right)$$

and an additive constant.

6 Frequently Asked Questions

This section is more discursive than the previous ones; "frequently" in its title means "at least once" (but with a reasonable expectation that a typical reader might well ask similar questions).

What is the role of the requirement of propriety in Theorem 1?

The theorem says that the log loss function leads to the most restrictive notion of randomness: if a sequence is random with respect to some prediction algorithm under the log loss function, then it is random with respect to the "same" prediction algorithm under an arbitrary CPM loss function. One should explain, however, what is meant by the same prediction algorithm, because of the freedom in parameterization (say, we can replace each prediction p by p^2). The requirement of propriety imposes a canonical parameterization.

What is the role of the requirement of mixability in Theorem 1?

The requirement of mixability ensures the existence of predictive complexity, which is used in the definition of predictive randomness.

Mixability is sufficient for the existence of predictive complexity (for computable loss functions). Is it also necessary?

Yes, it is: see Theorem 1 in [9].

What is the geometric intuition behind the notions of propriety and mixability?

The intuitions behind the two notions overlap; both involve requirements of convexity of the "superprediction set" (the area Northeast of the prediction set (2)). Let us suppose that the loss function λ is continuous in the prediction p, so that the prediction set is a curve. Propriety then means that the superprediction set is strictly convex (in particular, the prediction set has no straight segments) and that the points on the prediction set are indexed in a canonical way (namely, each such point is indexed by $1/(1-s)$ where $s<0$ is the slope of the tangent line to the prediction set at that point: cf. (12)). Mixability means that the superprediction set is convex in a stronger sense: it stays convex after being transformed by the mapping $(x,y)\in[0,\infty]^2\mapsto(e^{-\eta x},e^{-\eta y})$ for some $\eta>0$.

Why should we consider not only the log loss function (which nicely corresponds to probability distributions) but also other loss functions? You say "the log loss function, being most selective, should be preferred to the alternatives such as Brier or spherical loss". But this does not explain why these other loss functions were interesting in the first place.

Loss functions different from the log loss function are widely used in practice; in particular, the Brier loss function is at least as popular as (and perhaps even more popular than) the log loss function in machine learning: see, e.g., the extensive empirical study [2]. An important reason for the popularity of Brier loss is that the log loss function often leads to infinite average losses on large test sets for state-of-the-art prediction algorithms, which is considered to be "unfair", and some researchers even believe that any reasonable loss function should be bounded.

7 Conclusion

This note offers an answer to the problem of choosing a loss function for evaluating probabilistic prediction algorithms in experimental machine learning. Our answer is that the log loss function, being most selective, should be preferred to the alternatives such as Brier or spherical loss. This answer, however, remains asymptotic (involving unspecified constants) and raises further questions. To make it really practical, we need to restrict our generalized theory of algorithmic randomness, as Yuri did in a different context.

Acknowledgments.

I am grateful to Mitya Adamskiy, Yuri Kalnishkan, Ilia Nouretdinov, Ivan Petej, and Vladimir V'yugin for useful discussions. Thanks to an anonymous reviewer whose remarks prompted me to add Section 6 (and were used in both questions and answers). This work has been supported by EPSRC (grant EP/K033344/1) and the Air Force Office of Scientific Research (grant "Semantic Completions").

References

[1] J. Eric Bickel. Some comparisons among quadratic, spherical, and logarithmic scoring rules. *Decision Analysis*, 4:49–65, 2007.

[2] Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *Proceedings of the Twenty Third International Conference on Machine Learning*, pages 161–168, New York, 2006. ACM.

[3] David L. Dowe. Foreword re C. S. Wallace. *Computer Journal*, 51:523–560, 2008.

[4] David L. Dowe. Introduction to Ray Solomonoff 85th Memorial Conference. In *Algorithmic Probability and Friends: Bayesian Prediction and Artificial Intelligence*, volume 7070 of *Lecture Notes in Artificial Intelligence*, pages 1–36, Berlin, 2013. Springer.

[5] Yuri Gurevich and Grant O. Passmore. Impugning randomness, convincingly. *Studia Logica*, 82:1–31, 2012.

[6] David Haussler, Jyrki Kivinen, and Manfred K. Warmuth. Sequential prediction of individual sequences under general loss functions. *IEEE Transactions on Information Theory*, 44:1906–1925, 1998.

[7] Yuri Kalnishkan. Predictive complexity for games with finite outcome spaces. In Vladimir Vovk, Harris Papadopoulos, and Alex Gammerman, editors, *Measures of Complexity: Festschrift for Alexey Chervonenkis*, pages 119–141. Springer, Berlin, 2015. Available on the web.

[8] Yuri Kalnishkan and Vladimir Vovk. The existence of predictive complexity and the Legendre transformation. Technical Report CLRC-TR-00-04, Computer Learning Research Centre, Royal Holloway, University of London, March 2000.

[9] Yuri Kalnishkan, Vladimir Vovk, and Michael V. Vyugin. A criterion for the existence of predictive complexity for binary games. In Shai Ben-David, John Case, and Akira Maruoka, editors, *Proceedings of the Fifteenth International Conference on Algorithmic Learning Theory*, volume 3244 of *Lecture Notes in Computer Science*, pages 249–263, Heidelberg, 2004. Springer.

Full version: Technical Report CLRC-TR-04-04, Computer Learning Research Centre, Royal Holloway, University of London, 2004.

[10] Mark D. Reid, Rafael M. Frongillo, Robert C. Williamson, and Nishant Mehta. Generalized mixability via entropic duality. Technical Report `arXiv:1406.6130v1` [cs.LG], `arXiv.org` e-Print archive, June 2014. To appear in COLT 2015.

[11] Vladimir Vovk. Aggregating strategies. In Mark Fulk and John Case, editors, *Proceedings of the Third Annual Workshop on Computational Learning Theory*, pages 371–383, San Mateo, CA, 1990. Morgan Kaufmann.

[12] Vladimir Vovk. A game of prediction with expert advice. *Journal of Computer and System Sciences*, 56:153–173, 1998.

[13] Vladimir Vovk. Probability theory for the Brier game. *Theoretical Computer Science*, 261:57–79, 2001.