Conformal predictive distributions with kernels

Vladimir Vovk, Ilia Nouretdinov,
Valery Manokhin, and Alex Gammerman

практические выводы
теории вероятностей
могут быть обоснованы
в качестве следствий
гипотез о *предельной*
при данных ограничениях
сложности изучаемых явлений

On-line Compression Modelling Project (New Series)

Working Paper #20

First posted October 19, 2017. Last revised January 19, 2019.

Project web site:
http://alrw.net

Abstract

This paper reviews the checkered history of predictive distributions in statistics and discusses two developments, one from recent literature and the other new. The first development is bringing predictive distributions into machine learning, whose early development was so deeply influenced by two remarkable groups at the Institute of Automation and Remote Control. As result, they become more robust and their validity ceases to depend on Bayesian or narrow parametric assumptions. The second development is combining predictive distributions with kernel methods, which were originated by one of those groups, including Emmanuel Braverman. As result, they become more flexible and, therefore, their predictive efficiency improves significantly for realistic non-linear data sets.

This paper has been prepared for the Proceedings of the Braverman Readings, held in Boston on 28–30 April 2017.

Contents

1 Introduction

Prediction is a fundamental and difficult scientific problem. We limit the scope of our discussion by imposing, from the outset, two restrictions: we only want to predict one real number $y \in \mathbb{R}$, and we want our prediction to satisfy a reasonable property of validity (under a natural assumption). It can be argued that the fullest prediction for y is a probability measure on $\mathbb{R}$, which can be represented by its distribution function: see, e.g., [5, 6, 8]. We will refer to it as the predictive distribution. A standard property of validity for predictive distributions is being well-calibrated. Calibration can be defined as the "statistical compatibility between the probabilistic forecasts and the realizations" [8, Section 1.2], and its rough interpretation is that predictive distributions should tell the truth. Of course, truth can be uninteresting and non-informative, and there is a further requirement of efficiency, which is often referred to as sharpness [8, Section 2.3]. Our goal is to optimize the efficiency subject to validity [8, Section 1.2].

This paper is a very selective review of predictive distributions with validity guarantees. After introducing our notation and setting the prediction problem in Section 2, we start, in Section 3, from the oldest approach to predictive distributions, Bayesian. This approach gives a perfect solution but under a very restrictive assumption: we need a full knowledge of the stochastic mechanism generating the data. In Section 4 we move to Fisher's fiducial predictive distributions.

The first recent development (in [27], as described in Section 5 of this paper) was to carry over predictive distributions to the framework of statistical machine learning as developed by two groups at the Institute of Automation and Remote Control (Aizerman's laboratory including Braverman and Rozonoer and Lerner's laboratory including Vapnik and Chervonenkis; for a brief history of the Institute and research on statistical learning there, including the role of Emmanuel Markovich Braverman, see [25], especially Chapter 5). That development consisted in adapting predictive distributions to the IID model, discussed in detail in the next section. The simplest linear case was considered in [27], with groundwork laid in [1].

The second development, which is this paper's contribution, is combination with kernel methods, developed by the members of Aizerman's laboratory, first of all Braverman and Rozonoer [25, p. 48]; namely, in Section 6 we derive the kernelized versions of the main algorithms of [27]. In the experimental section (Section 8), we demonstrate an important advantage of kernelized versions. The computational efficiency of our methods is studied theoretically in Section 6, where we show that pre-processing a training sequence of length n takes, asymptotically, the same time as inverting an $n \times n$ matrix (at most n^3) and, after that, processing a test object takes time $O(n^2)$. Their predictive efficiency is studied in Section 8 experimentally using an artificial data set, where we show that a universal (Laplacian) kernel works remarkably well.

The standard methods of probabilistic prediction that have been used so far in machine learning, such as those proposed by Platt [15] and Zadrozny and

Elkan [29], are outside the scope of this paper for two reasons: first, they have no validity guarantees whatsoever, and second, they are applicable to classification problems, whereas in this paper we are interested in regression. A sister method to conformal prediction, Venn prediction, does have validity guarantees akin to those in conformal prediction (see, e.g., [26, Theorems 1 and 2]), but it is also applicable only to classification problems. Conformalized kernel ridge regression, albeit in the form of prediction intervals rather than predictive distributions, has been studied by Burnaev and Nazarov [2].

2 The problem

In this section we will introduce our basic prediction problem. The training sequence consists of n observations $z_i = (x_i, y_i) \in \mathbf{X} \times \mathbf{Y} = \mathbf{X} \times \mathbb{R}$, $i = 1, \ldots, n$; given a test object x_{n+1} we are asked to predict its label y_{n+1}. Each observation $z_i = (x_i, y_i)$, $i = 1, \ldots, n+1$, consists of two components, the object x_i assumed to belong to a measurable space $\mathbf{X}$ that we call the *object space* and the label y_i that belongs to a measurable space $\mathbf{Y}$ that we call the *label space*. In this paper we are interested in the case of regression, where the object space is the real line, $\mathbf{Y} = \mathbb{R}$.

In the problem of probability forecasting our prediction takes the form of a probability measure on the label space $\mathbf{Y}$; since $\mathbf{Y} = \mathbb{R}$, this measure can be represented by its distribution function. This paper is be devoted to this problem and its modifications.

Our prediction problem can be tackled under different assumptions. In the chronological order, the standard assumptions are Bayesian (discussed in Section 3 below), statistical parametric (discussed in Section 4), and nonparametric, especially the IID model, standard in machine learning (and discussed in detail in the rest of this section and further sections). When using the method of conformal prediction, it becomes convenient to differentiate between two kinds of assumptions, hard and soft (to use the terminology of [24]). Our hard assumption is the IID model: the observations are generated independently from the same probability distribution. The validity of our probabilistic forecasts will depend only on the hard model. In designing prediction algorithms, we may also use, formally or informally, another model in hope that it will be not too far from being correct and under which we optimize efficiency. Whereas the hard model is a standard statistical model (the IID model in this paper), the soft model is not always even formalized; a typical soft model (avoided in this paper) is the assumption that the label y of an object x depends on x in an approximately linear fashion.

In the rest of this paper we will use a fixed parameter $a > 0$, determining the amount of regularization that we wish to apply to our solution to the problem of prediction. Regularization becomes indispensable when kernel methods are used.

3 Bayesian solution

A very satisfactory solution to our prediction problem (and plethora of other problems of prediction and inference) is given by the theory that dominated statistical inference for more than 150 years, from the work of Thomas Bayes and Pierre-Simon Laplace to that of Karl Pearson, roughly from 1770 to 1930. This theory, however, requires rather strong assumptions.

Let us assume that our statistical model is linear in a feature space (спрямляемое пространство, in the terminology of Braverman and his colleagues) and the noise is Gaussian. Namely, we assume that $x_1, \dots, x_{n+1}$ is a deterministic sequence of objects and that the labels are generated as

$$y_i = w \cdot F(x_i) + \xi_i, \quad i = 1, \dots, n+1, \tag{1}$$

where $F : \mathbf{X} \to H$ is a mapping from the object space to a Hilbert space H, "$\cdot$" is the dot product in H, w is a random vector distributed as $N(0, (\sigma^2/a)I)$ (I being the identity operator on H), and ξ_i are random variables distributed as $N(0, \sigma^2)$ and independent of w and between themselves. Here a is the regularization constant introduced at the end of Section 2, and $\sigma > 0$ is another parameter, the standard deviation of the noise variables ξ_i.

It is easy to check that

$$\begin{aligned} \mathbb{E}\, y_i &= 0, & i &= 1, \dots, n, \\ \operatorname{cov}(y_i, y_j) &= \frac{\sigma^2}{a}\mathcal{K}(x_i, x_j) + \sigma^2 1_{\{i=j\}}, & i, j &= 1, \dots, n, \end{aligned} \tag{2}$$

where $\mathcal{K}(x, x') := F(x) \cdot F(x')$. By the theorem on normal correlation (see, e.g., [18, Theorem II.13.2]), the Bayesian predictive distribution for y_{n+1} given x_{n+1} and the training sequence is

$$N\left(k'(K + aI)^{-1}Y, \frac{\sigma^2}{a}\kappa + \sigma^2 - \frac{\sigma^2}{a}k'(K + aI)^{-1}k\right), \tag{3}$$

where k is the n-vector $k_i := \mathcal{K}(x_i, x_{n+1})$, $i = 1, \dots, n$, K is the kernel matrix for the first n observations (the training observations only), $K_{i,j} := \mathcal{K}(x_i, x_j)$, $i, j = 1, \dots, n$, $I = I_n$ is the $n \times n$ unit matrix, $Y := (y_1, \dots, y_n)'$ is the vector of the n training labels, and $\kappa := \mathcal{K}(x_{n+1}, x_{n+1})$.

The weakness of the model (1) (used, e.g., in [23, Section 10.3]) is that the Gaussian measure $N(0, (\sigma^2/a)I)$ exists only when H is finite-dimensional, but we can circumvent this difficulty by using (2) directly as our Bayesian model, for a given symmetric positive semidefinite $\mathcal{K}$. The mapping F in not part of the picture any longer. This is the standard approach in Gaussian process regression in machine learning.

In the Bayesian solution, there is no difference between the hard and soft model; in particular, (2) is required for the validity of the predictive distribution (3).

4 Fiducial predictive distributions

After its sesquicentennial rule, Bayesian statistics was challenged by Fisher and Neyman, who had little sympathy with each other's views apart from their common disdain for Bayesian methods. Fisher's approach was more ambitious, and his goal was to compute a full probability distribution for a future value (test label in our context) or for the value of a parameter. Neyman and his followers were content with computing intervals for future values (prediction intervals) and values of a parameter (confidence intervals).

Fisher and Neyman relaxed the assumptions of Bayesian statistics by allowing uncertainty, in Knight's [11] terminology. In Bayesian statistics we have an overall probability measure, i.e., we are in a situation of risk without any uncertainty. Fisher and Neyman worked in the framework of parametric statistics, in which we do not have any stochastic model for the value of the parameter (a number or an element of a Euclidean space). In the next section we will discuss the next step, in which the amount of uncertainty (where we lack a stochastic model) is even greater: our statistical model will be the nonparametric IID model (standard in machine learning).

The available properties of validity naturally become weaker as we weaken our assumptions. For predicting future values, conformal prediction (to be discussed in the next section) ensures calibration in probability, in the terminology of [8, Definition 1]. It can be shown that Bayesian prediction satisfies a stronger conditional version of this property: Bayesian predictive distributions are calibrated in probability conditionally on the training sequence and test object (more generally, on the past). The property of being calibrated in probability for conformal prediction is, on the other hand, unconditional; or, in other words, it is conditional on the trivial σ-algebra. Fisher's fiducial predictive distributions satisfy an intermediate property of validity: they are calibrated in probability conditionally on what was called the σ-algebra of invariant events in [13], which is greater than the trivial σ-algebra but smaller than the σ-algebra representing the full knowledge of the past. Our plan is to give precise statements with proofs in future work.

Fisher did not formalize his fiducial inference, and it has often been regarded as erroneous (his "biggest blunder" [7]). Neyman's simplification, replacing probability distributions by intervals, allowed him to state suitable notions of validity more easily, and his approach to statistics became mainstream until the Bayesian approach started to reassert itself towards the end of the 20th century. However, there has been a recent revival of interest in fiducial inference: cf. the BFF (Bayesian, frequentist, and fiducial) series of workshops, with the fourth one held on 1–3 May 2017 in Cambridge, MA, right after the Braverman Readings in Boston. Fiducial inference is a key topic of the series, both in the form of confidence distributions (the term introduced by David Cox [4] in 1958 for distributions for parameters) and predictive distributions (which by definition [17, Definition 1] must be calibrated in probability).

Since fiducial inference was developed in the context of parametric statistics, it has two versions, one targeting computing confidence distributions and the other predictive distributions. Under nonparametric assumptions, such as our IID model, we are not interested in confidence distributions (the parameter space, the set of all probability measures on the observation space $\mathbf{X}\times\mathbb{R}$, is just too big), and concentrate on predictive distributions. The standard notion of validity for predictive distributions, introduced independently by Schweder and Hjort [16, Chapter 12] and Shen, Liu, and Xie [17], is calibration in probability, going back to Philip Dawid's work (see, e.g., [5, Section 5.3] and [6]).

5 Conformal predictive distributions

In order to obtain valid predictive distributions under the IID model, we will need to relax slightly the notion of a predictive distribution as given in [17]. In our definition we will follow [27] and [22]; see those papers for further intuition and motivation.

Let $U = U[0,1]$ be the uniform probability distribution on the interval $[0,1]$. We fix the length n of the training sequence. Set $\mathbf{Z} := \mathbf{X}\times\mathbb{R}$; this is our *observation space*.

A function $Q : \mathbf{Z}^{n+1}\times[0,1]\to[0,1]$ is a *randomized predictive system* (RPS) if:

R1a For each training sequence $(z_1,\dots,z_n)\in\mathbf{Z}^n$ and each test object $x_{n+1}\in\mathbf{X}$, the function $Q(z_1,\dots,z_n,(x_{n+1},y),\tau)$ is monotonically increasing in both y and τ.

R1b For each training sequence $(z_1,\dots,z_n)\in\mathbf{Z}^n$ and each test object $x_{n+1}\in\mathbf{X}$,

$$\lim_{y\to-\infty} Q(z_1,\dots,z_n,(x_{n+1},y),0) = 0,$$
$$\lim_{y\to\infty} Q(z_1,\dots,z_n,(x_{n+1},y),1) = 1.$$

R2 For any probability measure P on $\mathbf{Z}$, $Q(z_1,\dots,z_n,z_{n+1},\tau)\sim U$ when $(z_1,\dots,z_{n+1},\tau)\sim P^{n+1}\times U$.

The function

$$Q_n : (y,\tau)\in\mathbb{R}\times[0,1]\mapsto Q(z_1,\dots,z_n,(x_{n+1},y),\tau) \qquad (4)$$

is the *randomized predictive distribution (function)* (RPD) output by the randomized predictive system Q on a training sequence $z_1,\dots,z_n$ and a test object x_{n+1}.

A *conformity measure* is a measurable function $A : \mathbf{Z}^{n+1}\to\mathbb{R}$ that is invariant with respect to permutations of the first n observations. A simple example, used in this paper, is

$$A(z_1,\dots,z_{n+1}) := y_{n+1} - \hat{y}_{n+1}, \qquad (5)$$

$\hat{y}_{n+1}$ being the prediction for y_{n+1} computed from x_{n+1} and $z_1, \ldots, z_{n+1}$ as training sequence. The *conformal transducer* determined by a conformity measure A is defined as

$$Q(z_1, \ldots, z_n, (x_{n+1}, y), \tau) := \frac{1}{n+1}\Bigl(\left|\left\{i = 1, \ldots, n+1 \mid \alpha_i^y < \alpha_{n+1}^y\right\}\right| \\ + \tau \left|\left\{i = 1, \ldots, n+1 \mid \alpha_i^y = \alpha_{n+1}^y\right\}\right|\Bigr), \quad (6)$$

where $(z_1, \ldots, z_n) \in \mathbf{Z}^n$ is a training sequence, $x_{n+1} \in \mathbf{X}$ is a test object, and for each $y \in \mathbb{R}$ the corresponding *conformity scores* α_i^y are defined by

$$\begin{aligned} \alpha_i^y &:= A(z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n, (x_{n+1}, y), z_i), \qquad i = 1, \ldots, n, \\ \alpha_{n+1}^y &:= A(z_1, \ldots, z_n, (x_{n+1}, y)). \end{aligned} \quad (7)$$

A function is a *conformal transducer* if it is the conformal transducer determined by some conformity measure. A *conformal predictive system* (CPS) is a function which is both a conformal transducer and a randomized predictive system. A *conformal predictive distribution* (CPD) is a function Q_n defined by (4) for a conformal predictive system Q.

The following lemma, stated in [27], gives simple conditions for a conformal transducer to be an RPS; it uses the notation of (7).

Lemma 1. *The conformal transducer determined by a conformity measure A is an RPS if, for each training sequence $(z_1, \ldots, z_n) \in \mathbf{Z}^n$, each test object $x_{n+1} \in \mathbf{X}$, and each $i \in \{1, \ldots, n\}$:*

- $\alpha_{n+1}^y - \alpha_i^y$ *is a monotonically increasing function of $y \in \mathbb{R}$;*
- $\lim_{y \to \pm\infty} \left(\alpha_{n+1}^y - \alpha_i^y\right) = \pm\infty$.

6 Kernel Ridge Regression Prediction Machine

In this section we introduce the Kernel Ridge Regression Prediction Machine (KRRPM); it will be the conformal transducer determined by a conformity measure of the form (5), where $\hat{y}_{n+1}$ is computed using kernel ridge regression, to be defined momentarily. There are three natural versions of the definition, and we start from reviewing them. All three versions are based on (1) as soft model (with the IID model being the hard model).

Given a training sequence $(z_1, \ldots, z_n) \in \mathbf{Z}^n$ and a test object $x_{n+1} \in \mathbf{X}$, the *kernel ridge regression* predicts

$$\hat{y}_{n+1} := k'(K + aI)^{-1}Y$$

for the label y_{n+1} of x_{n+1}. This is just the mean in (3), and the variance is ignored. Plugging this definition into (5), we obtain the *deleted KRRPM*. Alternatively, we can replace the conformity measure (5) by

$$A(z_1, \ldots, z_{n+1}) := y_{n+1} - \widehat{y}_{n+1}, \quad (8)$$

where

$$\widehat{y}_{n+1} := \bar{k}'(\bar{K} + aI)^{-1}\bar{Y} \tag{9}$$

is the prediction for the label y_{n+1} of x_{n+1} computed using $z_1, \ldots, z_{n+1}$ as the training sequence. The notation used in (9) is: $\bar{k}$ is the $(n+1)$-vector $k_i := \mathcal{K}(x_i, x_{n+1})$, $i = 1, \ldots, n+1$, $\bar{K}$ is the kernel matrix for all $n+1$ observations, $\bar{K}_{i,j} := \mathcal{K}(x_i, x_j)$, $i, j = 1, \ldots, n+1$, $I = I_{n+1}$ is the $(n+1) \times (n+1)$ unit matrix, and $\bar{Y} := (y_1, \ldots, y_{n+1})'$ is the vector of all $n+1$ labels. In this context, $\mathcal{K}$ is any given *kernel*, i.e., symmetric positive semidefinite function $\mathcal{K} : \mathbf{X}^2 \to \mathbb{R}$. The corresponding conformal transducer is the *ordinary KRRPM*. The disadvantage of the deleted and ordinary KRRPM is that they are not RPSs (they can fail to produce a function increasing in y in the presence of extremely high-leverage objects).

Set

$$\bar{H} := (\bar{K} + aI)^{-1}\bar{K} = \bar{K}(\bar{K} + aI)^{-1}. \tag{10}$$

This *hat matrix* "puts hats on the ys": according to (9), $\bar{H}\bar{Y}$ is the vector $(\widehat{y}_1, \ldots, \widehat{y}_{n+1})'$, where $\widehat{y}_i$, $i = 1, \ldots, n+1$, is the prediction for the label y_i of x_i computed using $z_1, \ldots, z_{n+1}$ as the training sequence. We will refer to the entries of the matrix $\bar{H}$ as $\bar{h}_{i,j}$ (where i is the row and j is the column of the entry), abbreviating $\bar{h}_{i,i}$ to $\bar{h}_i$. The usual relation between the residuals in (5) and (8) is

$$y_{n+1} - \hat{y}_{n+1} = \frac{y_{n+1} - \widehat{y}_{n+1}}{1 - \bar{h}_{n+1}}. \tag{11}$$

This equality makes sense since the diagonal elements $\bar{h}_i$ of the hat matrix are always in the semi-open interval $[0, 1)$ (and so the numerator is non-zero); for details, see Appendix A. Equation (11) motivates using the *studentized residuals* $(y_{n+1} - \widehat{y}_{n+1})(1 - \bar{h}_{n+1})^{-1/2}$, which are half-way between the deleted residuals in (5) and the ordinary residuals in (8). (We ignore a factor in the usual definition of studentized residuals, as in [14, (4.8)], that does not affect the value (6) of the conformal transducer.) The conformal transducer determined by the corresponding conformity measure

$$A(z_1, \ldots, z_{n+1}) := \frac{y_{n+1} - \widehat{y}_{n+1}}{\sqrt{1 - \bar{h}_{n+1}}} \tag{12}$$

is the (studentized) *KRRPM*. Later in this section we will see that the KRRPM is an RPS. This is the main reason why this is the main version considered in this paper, with "studentized" usually omitted.

An explicit form of the KRRPM

According to (6), to compute the predictive distributions produced by the KRRPM (in its studentized version), we need to solve the equation $\alpha_i^y = \alpha_{n+1}^y$ (and the corresponding inequality $\alpha_i^y < \alpha_{n+1}^y$) for $i = 1, \ldots, n+1$. Combining

Algorithm 1 Kernel Ridge Regression Prediction Machine

```
Require: A training sequence $(x_i, y_i) \in \mathbf{X} \times \mathbb{R}$, $i = 1, \dots, n$.
Require: A test object $x_{n+1} \in \mathbf{X}$.
1: Define the hat matrix $\bar{H}$ by (10), $\bar{K}$ being the $(n+1)\times(n+1)$ kernel matrix.
2: for $i \in \{1, 2, \dots, n\}$ do
3:     Define $A_i$ and $B_i$ by (13) and (14), respectively.
4:     Set $C_i := A_i / B_i$.
5: end for
6: Sort $C_1, \dots, C_n$ in the increasing order obtaining $C_{(1)} \le \dots \le C_{(n)}$.
7: Return the following predictive distribution for $y_{n+1}$:
       $Q_n(y, \tau) := \begin{cases} \frac{i+\tau}{n+1} & \text{if } y \in (C_{(i)}, C_{(i+1)}) \text{ for } i \in \{0, 1, \dots, n\} \\ \frac{i'-1+\tau(i''-i'+2)}{n+1} & \text{if } y = C_{(i)} \text{ for } i \in \{1, \dots, n\}. \end{cases}$    (15)
```

the definition (7) of the conformity scores α_i^y, the definition (12) of the conformity measure, and the fact that the predictions $\widehat{y}_i$ can be obtained from $\bar{Y}$ by applying the hat matrix $\bar{H}$ (cf. (10)), we can rewrite $\alpha_i^y = \alpha_{n+1}^y$ as

$$\frac{y_i - \sum_{j=1}^n \bar{h}_{ij} y_j - \bar{h}_{i,n+1} y}{\sqrt{1 - \bar{h}_i}} = \frac{y - \sum_{j=1}^n \bar{h}_{n+1,j} y_j - \bar{h}_{n+1} y}{\sqrt{1 - \bar{h}_{n+1}}}.$$

This is a linear equation, $A_i = B_i y$, and solving it we obtain $y = C_i := A_i / B_i$, where

$$A_i := \frac{\sum_{j=1}^n \bar{h}_{n+1,j} y_j}{\sqrt{1 - \bar{h}_{n+1}}} + \frac{y_i - \sum_{j=1}^n \bar{h}_{ij} y_j}{\sqrt{1 - \bar{h}_i}}, \tag{13}$$

$$B_i := \sqrt{1 - \bar{h}_{n+1}} + \frac{\bar{h}_{i,n+1}}{\sqrt{1 - \bar{h}_i}}. \tag{14}$$

The following lemma, to be proved in Appendix A, allows us to compute (6) easily.

Lemma 2. *It is always true that $B_i > 0$.*

The lemma gives Algorithm 1 for computing the conformal predictive distribution (4). The notation i' and i'' used in line 7 is defined as $i' := \min\{j \mid C_{(j)} = C_{(i)}\}$ and $i'' := \max\{j \mid C_{(j)} = C_{(i)}\}$, to ensure that $Q_n(y, 0) = Q_n(y-, 0)$ and $Q_n(y, 1) = Q_n(y+, 1)$ at $y = C_{(i)}$; $C_{(0)}$ and $C_{(n+1)}$ are understood to be $-\infty$ and ∞, respectively. Notice that there is no need to apply Lemma 1 formally; Lemma 2 makes it obvious that the KRRPM is a CPS.

Algorithm 1 is not computationally efficient for a large test set, since the hat matrix $\bar{H}$ (cf. (10)) needs to be computed from scratch for each test object. To obtain a more efficient version, we use a standard formula for inverting partitioned matrices (see, e.g., [10, (8)] or [23, (2.44)]) to obtain

$$\bar{H} = (\bar{K} + aI)^{-1}\bar{K} = \begin{pmatrix} K + aI & k \\ k' & \kappa + a \end{pmatrix}^{-1} \begin{pmatrix} K & k \\ k' & \kappa \end{pmatrix}$$

$$= \begin{pmatrix} (K + aI)^{-1} + d(K + aI)^{-1}kk'(K + aI)^{-1} & -d(K + aI)^{-1}k \\ -dk'(K + aI)^{-1} & d \end{pmatrix} \begin{pmatrix} K & k \\ k' & \kappa \end{pmatrix}$$

$$= \left(\begin{matrix} H + d(K + aI)^{-1}kk'H - d(K + aI)^{-1}kk' \\ -dk'H + dk' \end{matrix} \right. \tag{16}$$

$$\left. \begin{matrix} (K + aI)^{-1}k + d(K + aI)^{-1}kk'(K + aI)^{-1}k - d\kappa(K + aI)^{-1}k \\ -dk'(K + aI)^{-1}k + d\kappa \end{matrix} \right) \tag{17}$$

$$= \begin{pmatrix} H + d(K + aI)^{-1}kk'(H - I) & d(I - H)k \\ dk'(I - H) & -dk'(K + aI)^{-1}k + d\kappa \end{pmatrix} \tag{18}$$

$$= \begin{pmatrix} H - ad(K + aI)^{-1}kk'(K + aI)^{-1} & ad(K + aI)^{-1}k \\ adk'(K + aI)^{-1} & d\kappa - dk'(K + aI)^{-1}k \end{pmatrix}, \tag{19}$$

where

$$d := \frac{1}{\kappa + a - k'(K + aI)^{-1}k} \tag{20}$$

(the denominator is positive by the theorem on normal correlation, already used in Section 3), the equality in line (18) follows from $\bar{H}$ being symmetric (which allows us to ignore the upper right block of the matrix (16)–(17)), and the equality in line (19) follows from

$$I - H = (K + aI)^{-1}(K + aI) - (K + aI)^{-1}K = a(K + aI)^{-1}.$$

We have been using the notation H for the training hat matrix

$$H = (K + aI)^{-1}K = K(K + aI)^{-1}. \tag{21}$$

Notice that the constant ad occurring in several places in (19) is between 0 and 1:

$$ad = \frac{a}{a + \kappa - k'(K + aI)^{-1}k} \in (0, 1] \tag{22}$$

(the fact that $\kappa - k'(K + aI)^{-1}k$ is nonnegative follows from the lower right entry $\bar{h}_{n+1}$ of the hat matrix (19) being nonnegative; the nonnegativity of the diagonal entries of hat matrices is discussed in Appendix A).

The important components in the expressions for A_i and B_i (cf. (13) and (14)) are, according to (19),

$$\begin{aligned} 1 - \bar{h}_{n+1} &= 1 + dk'(K + aI)^{-1}k - d\kappa = 1 + \frac{k'(K + aI)^{-1}k - \kappa}{\kappa + a - k'(K + aI)^{-1}k} \\ &= \frac{a}{\kappa + a - k'(K + aI)^{-1}k} = ad, \end{aligned} \tag{23}$$

$$\begin{aligned} 1 - \bar{h}_i &= 1 - h_i + ade'_i(K + aI)^{-1}kk'(K + aI)e_i \\ &= 1 - h_i + ad(e'_i(K + aI)^{-1}k)^2, \end{aligned} \tag{24}$$

where $h_i = h_{i,i}$ is the ith diagonal entry of the hat matrix (21) for the n training objects and e_i is the ith vector in the standard basis of $\mathbb{R}^n$ (so that the jth component of e_i is $1_{\{i=j\}}$ for $j = 1, \dots, n$). Let $\hat{y}_i := e_i' HY$ be the prediction for y_i computed from the training sequence $z_1, \dots, z_n$ and the test object x_i. Using (23) (but not using (24) for now), we can transform (13) and (14) as

$$\begin{aligned}
A_i &:= \frac{\sum_{j=1}^n \bar{h}_{n+1,j} y_j}{\sqrt{1 - \bar{h}_{n+1}}} + \frac{y_i - \sum_{j=1}^n \bar{h}_{ij} y_j}{\sqrt{1 - \bar{h}_i}} \\
&= (ad)^{-1/2} \sum_{j=1}^n ady_j k'(K + aI)^{-1} e_j \\
&\quad + \frac{y_i - \sum_{j=1}^n h_{ij} y_j + \sum_{j=1}^n ady_j e_i'(K + aI)^{-1} kk'(K + aI)^{-1} e_j}{\sqrt{1 - \bar{h}_i}} \\
&= (ad)^{1/2} k'(K + aI)^{-1} Y + \frac{y_i - \hat{y}_i + ade_i'(K + aI)^{-1} kk'(K + aI)^{-1} Y}{\sqrt{1 - \bar{h}_i}}, \\
&= \sqrt{ad} \hat{y}_{n+1} + \frac{y_i - \hat{y}_i + ad\hat{y}_{n+1} e_i'(K + aI)^{-1} k}{\sqrt{1 - \bar{h}_i}}, \qquad (25)
\end{aligned}$$

where $\hat{y}_{n+1}$ is the Bayesian prediction for y_{n+1} (cf. the expected value in (3)), and

$$B_i := \sqrt{1 - \bar{h}_{n+1}} + \frac{\bar{h}_{i,n+1}}{\sqrt{1 - \bar{h}_i}} = \sqrt{ad} + \frac{adk'(K + aI)^{-1} e_i}{\sqrt{1 - \bar{h}_i}}. \qquad (26)$$

Therefore, we can implement Algorithm 1 as follows. Preprocessing the training sequence takes time $O(n^3)$ (or faster if using, say, the Coppersmith–Winograd algorithm and its versions; we assume that the kernel $\mathcal{K}$ can be computed in time $O(1)$):

1. The $n \times n$ kernel matrix K can be computed in time $O(n^2)$.
2. The matrix $(K + aI)^{-1}$ can be computed in time $O(n^3)$.
3. The diagonal of the training hat matrix $H := (K + aI)^{-1} K$ can be computed in time $O(n^2)$.
4. All $\hat{y}_i$, $i = 1, \dots, n$, can be computed by $\hat{y} := HY = (K + aI)^{-1}(KY)$ in time $O(n^2)$ (even without knowing H).

Processing each test object x_{n+1} takes time $O(n^2)$:

1. Vector k and number κ (as defined after (3)) can be computed in time $O(n)$ and $O(1)$, respectively.
2. Vector $(K + aI)^{-1} k$ can be computed in time $O(n^2)$.
3. Number $k'(K + aI)^{-1} k$ can now be computed in time $O(n)$.

4. Number d defined by (20) can be computed in time $O(1)$.

5. For all $i = 1, \dots, n$, compute $1 - \bar{h}_i$ as (24), in time $O(n)$ overall (given the vector computed in 2).

6. Compute the number $\hat{y}_{n+1} := k'(K + aI)^{-1}Y$ in time $O(n)$ (given the vector computed in 2).

7. Finally, compute A_i and B_i for all $i = 1, \dots, n$ as per (25) and (26), set $C_i := A_i/B_i$, and output the predictive distribution (15). This takes time $O(n)$ except for sorting the C_i, which takes time $O(n \log n)$.

7 Limitation of the KRRPM

The KRRPM makes a significant step forward as compared to the LSPM of [27]: our soft model (1) is no longer linear in x_i. In fact, using a universal kernel (such as Laplacian in Section 8) allows the function $x \in \mathbf{X} \mapsto w \cdot F(x)$ to approximate any continuous function (arbitrarily well within any compact set in $\mathbf{X}$). However, since we are interested in predictive distributions rather than point predictions, using the soft model (1) still results in the KRRPM being restricted. In this section we discuss the nature of the restriction, using the ordinary KRRPM as a technical tool.

The Bayesian predictive distribution (3) is Gaussian and (as clear from (1) and from the bottom right entry of (19) being nonnegative) its variance is at least σ^2. We will see that the situation with the conformal distribution is not as bad, despite the remaining restriction. To understand the nature of the restriction it will be convenient to ignore the denominator in (12), i.e., to consider the ordinary KRRPM; the difference between the (studentized) KRRPM and ordinary KRRPM will be small in the absence of high-leverage objects (an example will be given in the next section). For the ordinary KRRPM we have, in place of (13) and (14),

$$\begin{aligned} A_i &:= \sum_{j=1}^{n} \bar{h}_{n+1,j} y_j + y_i - \sum_{j=1}^{n} \bar{h}_{i,j} y_j, \\ B_i &:= 1 - \bar{h}_{n+1} + \bar{h}_{i,n+1}. \end{aligned}$$

Therefore, (25) and (26) become

$$A_i = ad\hat{y}_{n+1} + y_i - \hat{y}_i + ad\hat{y}_{n+1} e_i'(K + aI)^{-1}k$$

and

$$B_i = ad + ade_i'(K + aI)^{-1}k,$$

respectively. For $C_i := A_i/B_i$ we now obtain

$$C_i = \hat{y}_{n+1} + \frac{y_i - \hat{y}_i}{ad + ade_i'(K + aI)^{-1}k}$$

$$= \hat{y}_{n+1} + \frac{\sigma^2_{\text{Bayes}}/\sigma^2}{1 + e'_i(K + aI)^{-1}k}(y_i - \hat{y}_i), \quad (27)$$

where $\hat{y}_{n+1}$ is, as before, the Bayesian prediction for y_{n+1}, and σ^2_{Bayes} is the variance of the Bayesian predictive distribution (3) (cf. (22)).

The second addend $e'_i(K + aI)^{-1}k$ in the denominator of (27) is the prediction for the label of the test object x_{n+1} in the situation where all training labels are 0 apart from the ith, which is 1. For a long training sequence we can expect it to be close to 0 (unless x_i or x_{n+1} are highly influential); therefore, we can expect the shape of the predictive distribution output by the ordinary KRRPM to be similar to the shape of the empirical distribution function of the residuals $y_i - \hat{y}_i$. In particular, this shape does not depend (or depends weakly) on the test object x_{n+1}. This lack of sensitivity of the predictive distribution to the test object prevents the conformal predictive distributions output by the KRRPM from being universally consistent in the sense of [22]. The shape of the predictive distribution can be arbitrary, not necessarily Gaussian (as in (3)), but it is fitted to all training residuals and not just the residuals for objects similar to the test object. One possible way to get universally consistent conformal predictive distributions would be to replace the right-hand side of (5) by $\hat{F}_{n+1}(y_{n+1})$, where $\hat{F}_{n+1}$ is the Bayesian predictive distribution for y_{n+1} computed from x_{n+1} and $z_1, \dots, z_{n+1}$ as training sequence for a sufficiently flexible Bayesian model (in any case, more flexible than our homoscedastic model (1)). This idea was referred to as de-Bayesing in [23, Section 4.2] and frequentizing in [28, Section 3]. However, modelling input-dependent (heteroscedastic) noise efficiently is a well-known difficult problem in Bayesian regression, including Gaussian process regression (see, e.g., [9, 12, 19]).

8 Experimental results

In the first part of this section we illustrate the main advantage of the KRRPM over the LSPM introduced in [27], its flexibility: for a suitable kernel, it gets the location of the predictive distribution right. In the second part, we illustrate the limitation discussed in the previous section: while the KRRPM adapts to the shape of the distribution of labels, the adaptation is not conditional on the test object. Both points will be demonstrated using artificial data sets.

In our first experiment we generate a training sequence of length 1000 from the model

$$y_i = w_1 \cos x_{i,1} + w_2 \cos x_{i,2} + w_3 \sin x_{i,1} + w_4 \sin x_{i,2} + \xi_i, \quad (28)$$

where $(w_1, w_2, w_3, w_4) \sim N(0, I_4)$ (I_4 being the unit 4×4 matrix), $(x_{i,1}, x_{i,2}) \sim U[-1, 1]^2$ ($U[-1, 1]$ being the uniform probability distribution on $[-1, 1]$), and $\xi_i \sim N(0, 1)$, all independent. This corresponds to the Bayesian ridge regression model with $a = \sigma = 1$. The true kernel is

$$\mathcal{K}((x_1, x_2), (x'_1, x'_2))$$

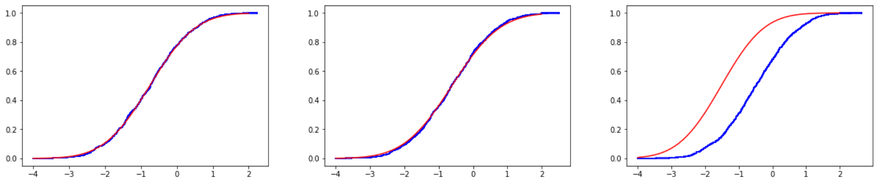

Figure 1: The predictive distribution for the label of the test object $(1, 1)$ based on a training sequence of length 1000 (all generated from the model (28)). The red line in each panel is the Bayesian predictive distribution based on the true kernel (29), and the blue line is the conformal predictive distribution based on: the true kernel (29) in the left-most panel; the Laplacian kernel in the middle panel; the linear kernel in the right-most panel.

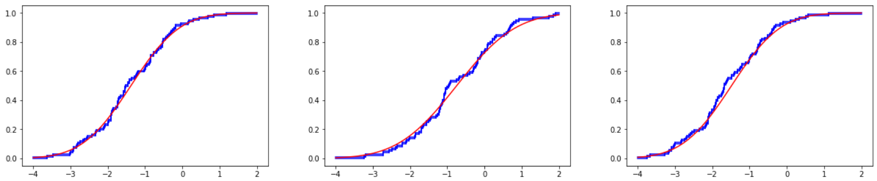

Figure 2: The analogue of Figure 1 for a training sequence of length 100.

$$
\begin{aligned}
&= (\cos x_1, \cos x_2, \sin x_1, \sin x_2) \cdot (\cos x'_1, \cos x'_2, \sin x'_1, \sin x'_2) \\
&= \cos(x_1 - x'_1) + \cos(x_2 - x'_2). \qquad (29)
\end{aligned}
$$

Remember that a kernel is *universal* [20] if any continuous function can be uniformly approximated (over each compact set) by functions in the corresponding reproducing kernel Hilbert space. An example of a universal kernel is the *Laplacian kernel*

$$\mathcal{K}(x, x') := \exp\left(-\left\|x - x'\right\|\right).$$

Laplacian kernels were introduced and studied in [21]; the corresponding reproducing kernel Hilbert space has the Sobolev norm

$$\left\|u\right\|^2 = 2\int_{-\infty}^{\infty} u(t)^2 \mathrm{d}t + 2\int_{-\infty}^{\infty} u'(t)^2 \mathrm{d}t$$

(see [21, Corollary 1]). This expression shows that Laplacian kernels are indeed universal. On the other hand, the *linear kernel* $\mathcal{K}(x, x') := x \cdot x'$ is far from being universal; remember that the LSPM [27] corresponds to this kernel and $a = 0$.

Figure 1 shows that, on this data set, universal kernels lead to better results. The parameter a in Figure 1 is the true one, $a = 1$. In the case of the Bayesian predictive distribution, the parameter $\sigma = 1$ is also the true one; remember that

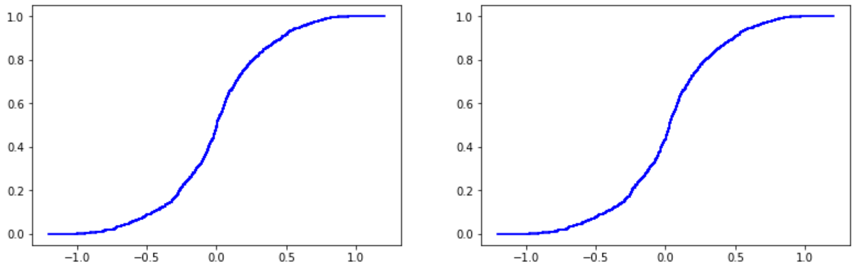

Figure 3: Left panel: predictions of the KRRPM for a training sequence of length 1000 and $x_{1001} = 0$. Right panel: predictions for $x_{1001} = 1$. The data are described in the text.

conformal predictive distributions do not require σ. The right-most panel shows that, when based on the linear kernel, the conformal predictive distribution can get the predictive distribution wrong. The other two panels show that the true kernel and, more importantly, the Laplacian kernel (chosen independently of the model (28)) are much more accurate. Figure 1 shows predictive distributions for a specific test object, $(1, 1)$, but this behaviour is typical. The effect of using a universal kernel becomes much less pronounced (or even disappears completely) for smaller lengths of the training sequence: see Figure 2 using 100 training observations (whereas Figure 1 uses 1000).

We now illustrate the limitation of the KRRPM that we discussed in the previous section. An artificial data set is generated as follows: $x_i \in [0, 1]$, $i = 1, \ldots, n$, are chosen independently from the uniform distribution U on $[0, 1]$, and $y_i \in [-x_i, x_i]$ are then chosen independently, again from the uniform distributions $U[-x_i, x_i]$ on their intervals. Figure 3 shows the prediction for $x_{n+1} = 0$ on the left and for $x_{n+1} = 1$ on the right for $n = 1000$; there is no visible difference between the studentized and ordinary versions of the KRRPM. The difference between the predictions for $x_{n+1} = 0$ and $x_{n+1} = 1$ is slight, whereas ideally we would like the former prediction to be concentrated at 0 whereas the latter should be close to the uniform distribution on $[-1, 1]$.

Fine details can be seen in Figure 4, which is analogous to Figure 3 but uses a training sequence of length $n = 10$. It shows the plots of the functions $Q_n(y, 0)$ and $Q_n(y, 1)$ of y, in the notation of (4). These functions carry all information about $Q_n(y, \tau)$ as function of y and τ since $Q_n(y, \tau)$ can be computed as the convex mixture $(1 - \tau)Q_n(y, 0) + \tau Q_n(y, 1)$ of $Q_n(y, 0)$ and $Q_n(y, 1)$.

In all experiments described in this section, the seed of the Python pseudorandom numbers generator was set to 0 for reproducibility. Our code can be found at http://alrw.net (Working Paper 20).

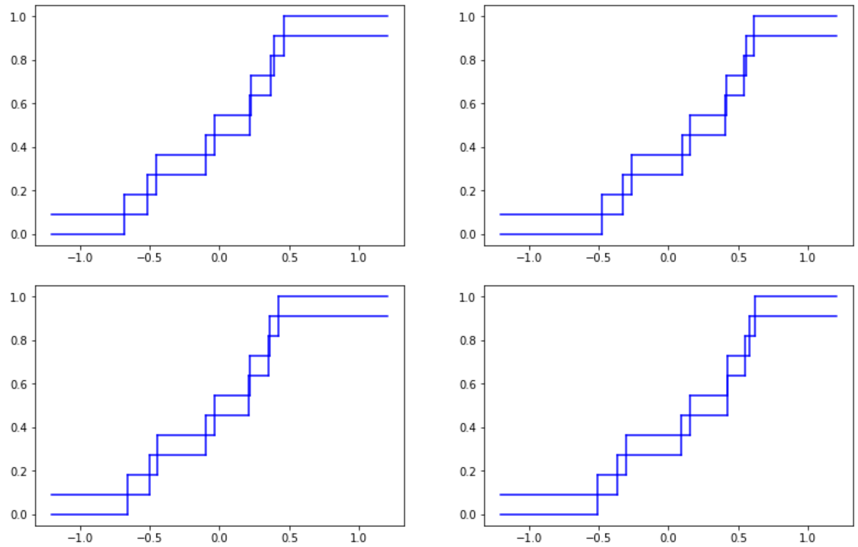

Figure 4: Upper left panel: predictions of the (studentized) KRRPM for a training sequence of length 10 and $x_{11} = 0$. Upper right panel: analogous predictions for $x_{11} = 1$. Lower left panel: predictions of the ordinary KRRPM for a training sequence of length 10 and $x_{11} = 0$. Lower right panel: analogous predictions for $x_{11} = 1$.

9 Conclusion

In this section we list some directions of further research:

- An important problem in practice is choosing a suitable value of the parameter a; it deserves careful study in the context of conformal predictive distributions.

- It was shown in [27] that, under narrow parametric statistical models, the LSPM is almost as efficient as various oracles that are optimal (or almost optimal) under those models; it would be interesting to prove similar results in the context of this paper using (1) as the model and the Bayesian predictive distribution (3) as the oracle.

- On the other hand, it would be interesting to explore systematically cases where (1) is violated and this results in poor performance of the Bayesian predictive distributions (cf. [23, Section 10.3, experimental results]). One example of such a situation is described in Section 7: in the case of non-Gaussian homogeneous noise, the Bayesian predictive distribution (3) is still Gaussian, whereas the KRRPM adapts to the noise distribution.

- To cope with heterogeneous noise distribution (see Section 7), we need to develop conformal predictive systems that are more flexible than the KRRPM.

Acknowledgments

This work has been supported by the EU Horizon 2020 Research and Innovation programme (in the framework of the ExCAPE project under grant agreement 671555) and Astra Zeneca (in the framework of the project "Machine Learning for Chemical Synthesis").

A Properties of the hat matrix

In the kernelized setting of this paper the hat matrix is defined as $H = (K + aI)^{-1}K$, where K is a symmetric positive semidefinite matrix whose size is denoted $n \times n$ in this appendix (cf. (10); in our current abstract setting we drop the bars over H and K and write n in place of $n + 1$). We will prove, or give references for, various properties of the hat matrix used in the main part of the paper.

Numerous useful properties of the hat matrix can be found in literature (see, e.g., [3]). However, the usual definition of the hat matrix is different from ours, since it is not kernelized; therefore, we start from reducing our kernelized definition to the standard one. Since K is symmetric positive semidefinite, it can be represented in the form $K = XX'$ for some matrix X, whose size will be denoted $n \times p$ (in fact, a matrix is symmetric positive semidefinite if and only if it can be represented as the Gram matrix of n vectors; this easily follows from the fact that a symmetric positive semidefinite K can be diagonalized: $K = Q'\Lambda Q$, where Q and Λ are $n \times n$ matrices, Λ is diagonal with nonnegative entries, and $Q'Q = I$). Now we can transform the hat matrix as

$$H = (K + aI)^{-1}K = (XX' + aI)^{-1}XX' = X(X'X + aI)^{-1}X'$$

(the last equality can be checked by multiplying both sides by $(XX' + aI)$ on the left). If we now extend X by adding $\sqrt{a}I_p$ on top of it (where $I_p = I$ is the $p \times p$ unit matrix),

$$\tilde{X} := \begin{pmatrix} \sqrt{a}I_p \\ X \end{pmatrix}, \tag{30}$$

and set

$$\tilde{H} := \tilde{X}(\tilde{X}'\tilde{X})^{-1}\tilde{X}' = \tilde{X}(X'X + aI)^{-1}\tilde{X}', \tag{31}$$

we will obtain a $(p + n) \times (p + n)$ matrix containing H in its lower right $n \times n$ corner. To find HY for a vector $Y \in \mathbb{R}^n$, we can extend Y to $\tilde{Y} \in \mathbb{R}^{p+n}$ by adding p zeros at the beginning of Y and then discard the first p elements in $\tilde{H}\tilde{Y}$. Notice that $\tilde{H}$ is the usual definition of the hat matrix associated with the data matrix $\tilde{X}$ (cf. [3, (1.4a)]).

When discussing (11), we used the fact that the diagonal elements of H are in $[0,1)$. It is well-known that the diagonal elements of the usual hat matrix, such as $\tilde{H}$, are in $[0,1]$ (see, e.g., [3, Property 2.5(a)]). Therefore, the diagonal elements of H are also in $[0,1]$. Let us check that h_i are in fact in the semi-open interval $[0,1)$ directly, without using the representation in terms of $\tilde{H}$. Representing $K = Q'\Lambda Q$ as above, where Λ is diagonal with nonnegative entries and $Q'Q = I$, we have

$$\begin{aligned} H = (K+aI)^{-1}K &= (Q'\Lambda Q + aI)^{-1}Q'\Lambda Q = (Q'(\Lambda + aI)Q)^{-1}Q'\Lambda Q \\ &= Q^{-1}(\Lambda + aI)^{-1}(Q')^{-1}Q'\Lambda Q = Q'(\Lambda + aI)^{-1}\Lambda Q. \end{aligned} \quad (32)$$

The matrix $(\Lambda + aI)^{-1}\Lambda$ is diagonal with the diagonal entries in the semi-open interval $[0,1)$. Since $Q'Q = I$, the columns of Q are vectors of length 1. By (32), each diagonal element of H is then of the form $\sum_{i=1}^n \lambda_i q_i^2$, where all $\lambda_i \in [0,1)$ and $\sum_{i=1}^n q_i^2 = 1$. We can see that each diagonal element of H is in $[0,1)$.

The equality (11) itself was used only for motivation, so we do not prove it; for a proof in the non-kernelized case, see, e.g., [14, (4.11) and Appendix C.7].

Proof of Lemma 2

In our proof of $B_i > 0$ we will assume $a > 0$, as usual. We will apply the results discussed so far in this appendix to the matrix $\bar{H}$ in place of H and to $n+1$ in place of n.

Our goal is to check the strict inequality

$$\sqrt{1-\bar{h}_{n+1}} + \frac{\bar{h}_{i,n+1}}{\sqrt{1-\bar{h}_i}} > 0; \quad (33)$$

remember that both $\bar{h}_{n+1}$ and $\bar{h}_i$ are numbers in the semi-open interval $[0,1)$. The inequality (33) can be rewritten as

$$\bar{h}_{i,n+1} > -\sqrt{(1-\bar{h}_{n+1})(1-\bar{h}_i)} \quad (34)$$

and in the weakened form

$$\bar{h}_{i,n+1} \geq -\sqrt{(1-\bar{h}_{n+1})(1-\bar{h}_i)} \quad (35)$$

follows from [3, Property 2.6(b)] (which can be applied to $\tilde{H}$).

Instead of the original hat matrix $\bar{H}$ we will consider the extended matrix (31), where $\tilde{X}$ is defined by (30) with $\bar{X}$ in place of X. The elements of $\tilde{H}$ will be denoted $\tilde{h}$ with suitable indices, which will run from $-p+1$ to $n+1$, in order to have the familiar indices for the submatrix $\bar{H}$. We will assume that we have an equality in (34) and arrive at a contradiction. There will still be an equality in (34) if we replace $\bar{h}$ by $\tilde{h}$, since $\tilde{H}$ contains $\bar{H}$. Consider auxiliary "random residuals" $E := (I - \tilde{H})\epsilon$, where ϵ is a standard Gaussian random vector in $\mathbb{R}^{p+n+1}$; there are $p+n+1$ random residuals $E_{-p+1},\ldots,E_{n+1}$. Since the correlation between the random residuals E_i and E_{n+1} is

$$\operatorname{corr}(E_i,E_{n+1}) = \frac{-\tilde{h}_{i,n+1}}{\sqrt{(1-\tilde{h}_{n+1})(1-\tilde{h}_i)}}$$

(this easily follows from $I-\tilde{H}$ being a projection matrix and is given in, e.g., [3, p. 11]), (35) is indeed true. Since we have an equality in (34) (with $\tilde{h}$ in place of $\bar{h}$), E_i and E_{n+1} are perfectly correlated. Remember that neither row number i nor row number $n+1$ of the matrix $I-\bar{H}$ are zero (since the diagonal elements of $\bar{H}$ are in the semi-open interval $[0,1)$), and so neither E_i nor E_{n+1} are zero vectors. Since E_i and E_{n+1} are perfectly correlated, the row number i of the matrix $I-\tilde{H}$ is equal to a positive scalar c times its row number $n+1$. The projection matrix $I-\tilde{H}$ then projects $\mathbb{R}^{p+n+1}$ onto a subspace of the hyperplane in $\mathbb{R}^{p+n+1}$ consisting of the points with coordinate number i being c times the coordinate number $n+1$. The orthogonal complement of this subspace, i.e., the range of $\tilde{H}$, will contain the vector $(0,\ldots,0,-1,0,\ldots,0,c)$ (-1 being its coordinate number i). Therefore, this vector will be in the range of $\tilde{X}$ (cf. (31)). Therefore, this vector will be a linear combination of the columns of the extended matrix (30) (with $\bar{X}$ in place of X), which is impossible because of the first p rows $\sqrt{a}I_p$ of the extended matrix.

References

[1] Evgeny Burnaev and Vladimir Vovk. Efficiency of conformalized ridge regression, On-line Compression Modelling project (New Series), `http://alrw.net`, Working Paper 10, February 2014.

[2] Evgeny V. Burnaev and Ivan N. Nazarov. Conformalized Kernel Ridge Regression. Technical Report arXiv:1609.05959 [stat.ML], arXiv.org e-Print archive, September 2016. Conference version: *Proceedings of the Fifteenth International Conference on Machine Learning and Applications (ICMLA 2016)*, pp. 45–52.

[3] Samprit Chatterjee and Ali S. Hadi. *Sensitivity Analysis in Linear Regression.* Wiley, New York, 1988.

[4] David R. Cox. Some problems connected with statistical inference. *Annals of Mathematical Statistics*, 29:357–372, 1958.

[5] A. Philip Dawid. Statistical theory: the prequential approach (with discussion). *Journal of the Royal Statistical Society A*, 147:278–292, 1984.

[6] A. Philip Dawid and Vladimir Vovk. Prequential probability: Principles and properties. *Bernoulli*, 5:125–162, 1999.

[7] Bradley Efron. R. A. Fisher in the 21st century. *Statistical Science*, 13:95–122, 1998.

[8] Tilmann Gneiting and Matthias Katzfuss. Probabilistic forecasting. *Annual Review of Statistics and Its Application*, 1:125–151, 2014.

[9] Paul W. Goldberg, Christopher K. I. Williams, and Christopher M. Bishop. Regression with input-dependent noise: A Gaussian process treatment. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 493–499, Cambridge, MA, 1998. MIT Press.

[10] Harold V. Henderson and Shayle R. Searle. On deriving the inverse of a sum of matrices. *SIAM Review*, 23:53–60, 1981.

[11] Frank H. Knight. *Risk, Uncertainty, and Profit.* Houghton Mifflin Company, Boston, MA, 1921.

[12] Quoc V. Le, Alex J. Smola, and Stéphane Canu. Heteroscedastic Gaussian process regression. In Rina Dechter and Thomas Richardson, editors, *Proceedings of the Twenty Second International Conference on Machine Learning*, pages 461–468, New York, 2005. ACM.

[13] Peter McCullagh, Vladimir Vovk, Ilia Nouretdinov, Dmitry Devetyarov, and Alex Gammerman. Conditional prediction intervals for linear regression. In *Proceedings of the Eighth International Conference on Machine Learning and Applications (ICMLA 2009)*, pages 131–138, 2009. Available from `http://www.stat.uchicago.edu/~pmcc/reports/predict.pdf`.

[14] Douglas C. Montgomery, Elizabeth A. Peck, and G. Geoffrey Vining. *Introduction to Linear Regression Analysis.* Wiley, Hoboken, NJ, fifth edition, 2012.

[15] John C. Platt. Probabilities for SV machines. In Alexander J. Smola, Peter L. Bartlett, Bernhard Schölkopf, and Dale Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 2000.

[16] Tore Schweder and Nils L. Hjort. *Confidence, Likelihood, Probability: Statistical Inference with Confidence Distributions.* Cambridge University Press, Cambridge, UK, 2016.

[17] Jieli Shen, Regina Liu, and Minge Xie. Prediction with confidence—a general framework for predictive inference. *Journal of Statistical Planning and Inference*, 195:126–140, 2018.

[18] Albert N. Shiryaev. Вероятность *(Probability).* МЦНМО, Moscow, third edition, 2004.

[19] Edward Snelson and Zoubin Ghahramani. Variable noise and dimensionality reduction for sparse Gaussian processes. In Rina Dechter and Thomas Richardson, editors, *Proceedings of the Twenty Second Conference on Uncertainty in Artifical Intelligence (UAI 2006)*, pages 461–468, Arlington, VA, 2006. AUAI Press.

[20] Ingo Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93, 2001.

[21] Christine Thomas-Agnan. Computing a family of reproducing kernels for statistical applications. *Numerical Algorithms*, 13:21–32, 1996.

[22] Vladimir Vovk. Universally consistent predictive distributions, On-line Compression Modelling project (New Series), `http://alrw.net`, Working Paper 18, September 2017.

[23] Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005.

[24] Vladimir Vovk, Ilia Nouretdinov, and Alex Gammerman. On-line predictive linear regression, On-line Compression Modelling project (New Series), `http://alrw.net`, Working Paper 1, November 2011.

[25] Vladimir Vovk, Harris Papadopoulos, and Alex Gammerman, editors. *Measures of Complexity: Festschrift for Alexey Chervonenkis*. Springer, Berlin, 2015.

[26] Vladimir Vovk and Ivan Petej. Venn–Abers predictors, On-line Compression Modelling project (New Series), `http://alrw.net`, Working Paper 7, April 2014 (first posted in October 2012).

[27] Vladimir Vovk, Jieli Shen, Valery Manokhin, and Minge Xie. Nonparametric predictive distributions based on conformal prediction, On-line Compression Modelling project (New Series), `http://alrw.net`, Working Paper 17, April 2017.

[28] Larry Wasserman. Frasian inference. *Statistical Science*, 26:322–325, 2011.

[29] Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In Carla E. Brodley and Andrea P. Danyluk, editors, *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001)*, pages 609–616, San Francisco, CA, 2001. Morgan Kaufmann.