Linear prediction and curve fitting: Gaussian-process emulation in theory and practice

Sylvy Anscombe
Université de Paris and Sorbonne Université,
CNRS, IMJ-PRG, F-75006 Paris, France
and
Amery Gration
Rudolf Peierls Centre for Theoretical Physics,
Clarendon Laboratory, Parks Road, Oxford, OX1 3PU, U.K.

December 12, 2022

Abstract

We consider linear prediction using only elementary linear algebra, and derive the well-known formulas for the best linear predictor, the best linear unbiased predictor, and the best linear unbiased estimator in this setting. Our approach is analogous to that used by Parzen (1959) but without the machinery of reproducing kernel Hilbert spaces. We also consider the application of linear prediction to the task of curve fitting, and extend the theory of leave-p-out cross-validation developed by Dubrule (1983) and Haslett & Hayes (1998) to include the best linear predictor and the best linear unbiased estimator.

Keywords: Best linear predictor; Best linear unbiased predictor; Kriging; Gaussian process; Cross-validation.

1 Introduction

In *linear prediction* we seek a predictor for a random variable Z that is a linear combination of the elements of a random process $X = (X_i)_i$. The *best linear predictor*, denoted $Z^*_{[X]}$, minimizes the *mean-square error* $\mathrm{E}((Z - Z^*_{[X]})^2)$. Remarkably, we may find the best linear predictor and its mean-square error even if we do not know the expectations $\mathrm{E}(Z)$ and $\mathrm{E}(X)$. However, without knowledge of these expectations, we may not minimize the mean-square error of a random variable subject to the constraint of unbiasedness. We therefore assume that E is an element of a given space of linear functionals F and define the *best linear unbiased predictor*, denoted $Z^\dagger_{[X]}$, to be the linear predictor that minimizes the mean-square error subject to the constraint that $f(Z) = f(Z^\dagger_{[X]})$ for all $f \in F$.

The standard way of finding the best linear predictor is by direct minimization of the mean-square error under the assumption that X is finite. Similarly, the standard way of finding the best linear unbiased predictor is by constrained minimization of the mean-square error using the method of Lagrangian multipliers, again under the assumption that X is finite (Goldberger 1962, Henderson 1963). Parzen (1959), however, gave a general theory of linear prediction in a geometric setting. In this general theory, X is permitted to be infinite and the predictor may be a convergent series in a Hilbert space of random variables. The best linear predictor of Z and the best linear unbiased predictor of Z are then the projections of Z onto the appropriate closed subspaces. In fact, Parzen works not in a Hilbert space of random variables, but in a reproducing kernel Hilbert space that is isomorphic to this Hilbert space of random variables (see also Wahba 2003). We will follow Parzen in spirit, but without the machinery of reproducing kernel Hilbert spaces. For the sake of clarity we will assume that X is finite and work directly in a linear vector space of random variables. In our framework linear estimation is a special case of prediction, and the best linear unbiased estimator of the mean of a random variable is a special case of the best linear unbiased predictor.

Once we have a method for constructing a predictor we may use it to predict the elements of a random process, $Z := (Z_i)_i$. This procedure seems not to have a name. Let us call it 'replication', and let us call a family of predictors for the elements of Z a 'replicator for Z based on X'. If every element of a replicator is a linear predictor we will call that

replicator a 'linear replicator'. An important application of linear replication is in curve fitting. We observe that any curve may be viewed as the realization of a random process, Z, and view X as a subset of this random process (possibly contaminated by noise). If we treat our data as a realization of X then we may compute the equivalent realization of the replicator, which serves as the fitted curve. In turn, an important application of curve fitting is in the construction of surrogate models for expensive computer simulations (Sacks et al. 1989). In this context, and under the further assumption that Z has a Gaussian distribution, it is often called 'Gaussian-process emulation' (Rasmussen & Williams 2006).

Although it is true that any curve may be viewed as the realization of a random process this observation in itself is of limited use. For an arbitrary curve we have no idea of *which* random process it is a realization. We must therefore make arbitrary assumptions about this random process, its mean, second moments and distribution. Consequently, we must be sure to test the performance of any replicator that we use for curve fitting. One way of doing this is *leave-p-out cross-validation*, in which we partition our sample, of size n, into two sets, one of size p and one of size $n - p$, and form predictors for the first set based on the second. We must ensure that the residuals of the predictors are consistent with our assumptions. In validating a replicator we may take advantage of a theory of linear prediction residuals developed by Dubrule (1983) and Haslett & Hayes (1998) that relates the leave-p-out best linear unbiased predictor residuals to the best linear unbiased estimator residuals. We show that, just as in our framework linear estimation is a special case of linear prediction, so in their framework the cross-validation of linear estimators is a special case of the cross-validation of linear predictors.

2 Linear prediction

We will work in an inner product space, $\mathbf{V} := (V, \langle \cdot, \cdot \rangle)$, consisting of a vector space, V, of second-order random variables on a common probability space, equipped with an inner product, $\langle \cdot, \cdot \rangle$, given by $\langle P, Q \rangle := \mathrm{E}(PQ)$.[1] We will use the notation 'mom' for the second moment function, such that $\mathrm{mom}(P, Q) := \mathrm{E}(PQ)$, in order to emphasize the analogy

[1] We will use upper-case letters to denote random variables and random processes, and lower-case letters to denote their realizations.

between the second moment function and the covariance function, cov. Note, however, that the covariance function itself is not an inner product on V since it is not point-separating. The inner product induces a norm, $\|\cdot\|$, on V, given by $\|P\| := \sqrt{\langle P, P\rangle}$, whereas the covariance induces a semi-norm, $\|\cdot\|_{\mathrm{cov}}$, on V, given by $\|P\|_{\mathrm{cov}} := \sqrt{\mathrm{cov}(P, P)}$. Given a random variable, Z, and a random process, $X = (X_i)_{i\leq n}$, a *linear predictor of Z based on X* is any element of $\mathrm{span}(X) \subseteq \mathbf{V}$. We quantify the performance of a linear predictor, Y, using the mean-square error, $\mathrm{MSE}(Y) = \mathrm{E}((Z-Y)^2) = \|Z-Y\|^2$, which we may think of as the squared distance between Y and Z.

Remark 1. The vector space V is not given a distinguished basis, so we do not identify an element of V with a vector (i.e. column vector) of real numbers. Nevertheless, elements of $\mathrm{span}(X)$ are linear combinations of the elements of X. An element of $\mathrm{span}(X)$ is equal to a product $a^{\mathrm{t}}X$, where $a \in \mathbb{R}^n$ is a real vector and we view $X \in V^n$ as a vector of random variables. Matrix notation is used here simply to represent summation. Given two random variables $a^{\mathrm{t}}X \in \mathrm{span}(X)$, where $a \in \mathbb{R}^n$, and $Y \in \mathbf{V}$, we have $\langle a^{\mathrm{t}}X, Y\rangle = a^{\mathrm{t}}(\langle X_i, Y\rangle)_{i\leq n}$.

2.1 Best linear predictor

Geometrically, the best linear predictor (BLP) of Z is the orthogonal projection of Z onto $\mathrm{span}(X)$, which we will write as $\pi_{\mathrm{span}(X)}(Z)$. It is clear from the definition that the BLP is unique, and that it exists because X is finite. We may write the orthogonal projection of Z onto $\mathrm{span}(X)$ using the following lemma.

Lemma 2. *Let $\mathbf{A} = (A, g)$ be a real vector space equipped with the positive-semidefinite symmetric bilinear form $g : A \times A \longrightarrow \mathbb{R}$. Let $\mathbf{B} \subseteq \mathbf{A}$ be a finite dimensional subspace with a basis $e = (e_i)_{i\leq n}$. Suppose that the restriction of g to B is positive definite. Then orthogonal projection with respect to g of $a \in \mathbf{A}$ onto $\mathbf{B}$ is the map*

$$\begin{aligned} \pi^g_{\mathbf{B}} : \mathbf{A} &\longrightarrow \mathbf{B} \\ a &\longmapsto (g(a, e_i))^{\mathrm{t}}_{i\leq n} G_e^{-1} e \end{aligned}$$

where $G_e = (g(e_i, e_j))_{i\leq n, j\leq n}$ is the Gram matrix of e.

Proof. A proof is given by Bourbaki (1981, EVT V.13). □

We may also write the norm of any element of $\mathrm{span}(X)$ using the following lemma.

Lemma 3. *Let $\mathbf{A} = (A, \langle\cdot,\cdot\rangle)$ be a real inner product space with basis $e = (e_i)_{i\leq n}$. Let $a \in \mathbf{A}$ be a vector, and suppose that a is represented by the column vector v with respect to e, so that $a = v^{\mathrm{t}}e$. Then $\|a\|^2 = v^{\mathrm{t}}G_e v$, where $G_e = (\langle e_i, e_j\rangle)_{i\leq n, j\leq n}$ is the Gram matrix of e.*

Proof. A proof is given by Lang (2002, XIII, C6). □

Note that the second-moment matrix of X, namely $R = (\langle X_i, X_j\rangle)_{i\leq n, j\leq n}$, is the Gram matrix of X. If X is linearly independent then R is invertible (as we assume henceforth) and the following proposition is an immediate consequence of Lemmas 2 and 3.

Proposition 4. *The BLP of Z based on $X = (X_i)_{i\leq n}$ is*

$$Z^*_{[X]} = \rho^{\mathrm{t}} R^{-1} X,$$

where $\rho = (\mathrm{mom}(Z, X_i))_{i\leq n}$. It has mean-square error

$$\mathrm{MSE}(Z^*_{[X]}) = \mathrm{mom}(Z, Z) - \rho^{\mathrm{t}} R^{-1} \rho.$$

Remark 5. It is worth emphasizing that the BLP is, in general, a biased predictor. Specifically, $\mathrm{bias}(Z^*_{[X]}) = \mathrm{E}(Z - Z^*_{[X]}) = \mathrm{E}(Z) - \rho^{\mathrm{t}} R^{-1} \mathrm{E}(X)$.

2.2 Best linear unbiased predictor

We now seek an unbiased linear predictor of Z. For this we will require a generalized notion of unbiasedness. Recall that if Y is an unbiased predictor for Z then $\mathrm{E}(Y) = \mathrm{E}(Z)$, and note that the expectation function, $\mathrm{E} : \mathbf{V} \longrightarrow \mathbb{R}$ is a continuous linear functional on $\mathbf{V}$, i.e. an element of the continuous dual space $\mathbf{V}'$. Let $F \subseteq \mathbf{V}'$ be a subspace of the continuous dual space. We will say that a linear predictor of Z, namely Y, is *F-unbiased* if $f(Y) = f(Z)$ for all $f \in F$. We will call any element of F a 'pseudoexpectation function' and we will call F itself the 'space of pseudoexpectation functions'. From now on, we will assume that the expectation function, E, is an element of F. Then any predictor, Y, that is F-unbiased is necessarily unbiased (i.e. E-unbiased).

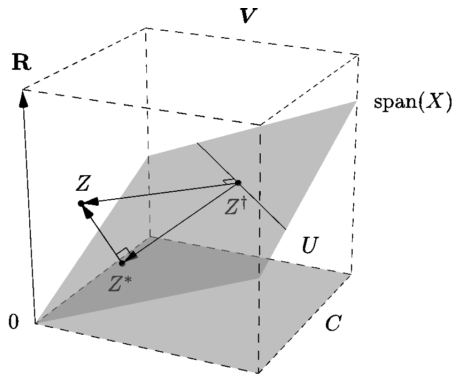

Figure 1: A schematic representation of $Z^*_{[X]}$, the best linear predictor of Z based on X (Prop. 4), and $Z^\dagger_{[X]}$, the best linear unbiased predictor of Z based on X (Prop. 6).

The *best linear unbiased predictor (BLUP) of Z based on X* is the linear predictor Y that is F-unbiased and minimizes the mean-square error among all F-unbiased linear predictors. We denote the BLUP of Z by $Z^\dagger_{[X]}$. The set of F-unbiased linear predictors of Z is in fact an affine subspace of V, namely

$$U := \{Y \in \mathrm{span}(X) \mid f(Y) = f(Z), \text{ for all } f \in F\}.$$

Geometrically, then, the BLUP of Z is the orthogonal projection of Z onto U, which we write as $\pi_U(Z)$.

The BLUP of Z is shown schematically in Figure 1. Note that the space $\mathbf{V}$ admits a decomposition as an orthogonal direct sum of the subspace, $\mathbf{C}$, of centred random variables and the subspace of constant random variables, which we identify with $\mathbb{R}$, i.e. $\mathbf{V} = \mathbf{C} \oplus \mathbb{R}$.[2] Because projections compose, the BLUP is not only the orthogonal projection of Z onto U but also the projection of the BLP onto U, i.e. $Z^\dagger_{[X]} = \pi_U(Z^*_{[X]})$. The BLUP does not necessarily exist. Indeed it exists if and only if U is not empty. However, when it exists it is unique.

We will require two assumptions: (1) that the restriction of the covariance function to $\mathrm{span}(X)$ is an inner product, which, for convenience, we will denote $\langle \cdot, \cdot \rangle_{\mathrm{cov}}$, such that $\langle P, Q \rangle_{\mathrm{cov}} = \mathrm{cov}(P, Q)$ for all $P, Q \in \mathrm{span}(X)$, and (2) that F is of finite-dimension and has a basis $f = (f_q)_{q \leq m}$. Given the first assumption, the covariance matrix of X, namely $K = (\langle X_i, X_j \rangle_{\mathrm{cov}})_{i \leq n, j \leq n}$, is in fact the Gram matrix of X, and is positive-definite, hence

[2]Note that the orthogonal projection of $\mathbf{V}$ onto $\mathbb{R}$ is the expectation.

invertible since a matrix is invertible if and only if it is positive definite. Given the second assumption, by the Riesz representation theorem there exists for each $f_q \in \mathbf{V}'$ a random variable $E_q \in \operatorname{span}(X)$ such that $\langle E_q, Y\rangle_{\mathrm{cov}} = f_q(Y)$ for all $Y \in \operatorname{span}(X)$. Then $(E_q)_{q\leq m}$ is a basis for the subspace $M \subseteq \operatorname{span}(X)$ that corresponds to F. We then find the following.

Proposition 6. *The BLUP of Z based on $X = (X_i)_{i\leq n}$ is*

$$Z^\dagger_{[X]} = \sigma^{\mathrm{t}} K^{-1} X - \sigma^t K^{-1}\Phi^{\mathrm{t}}(\Phi K^{-1}\Phi^{\mathrm{t}})^{-1}\Phi K^{-1} X + \varphi^{\mathrm{t}}(\Phi K^{-1}\Phi^{\mathrm{t}})^{-1}\Phi K^{-1} X. \tag{1}$$

where $\sigma = (\operatorname{cov}(Z, X_i))_{i\leq n}$, $\varphi = (f_q(Z))_{q\leq m}$, and $\Phi = (f_q(X_i))_{q\leq m, i\leq n}$. It has mean-square error

$$\operatorname{MSE}(Z^\dagger_{[X]}) = \operatorname{cov}(Z, Z) - \sigma^{\mathrm{t}} K^{-1}\sigma + (\varphi - \Phi K^{-1}\sigma)^{\mathrm{t}}(\Phi K^{-1}\Phi^{\mathrm{t}})^{-1}(\varphi - \Phi K^{-1}\sigma). \tag{2}$$

Proof. Define the *cov-orthogonal projection* of Z onto a subspace $S \subseteq V$ to be the element $\pi^{\mathrm{cov}}_S(Z)$ that minimizes the semi-norm $\|Z - \pi^{\mathrm{cov}}_S(Z)\|_{cov}$ among elements of S. For any $Y \in U$, we have

$$\|Z - Y\|^2_{\mathrm{cov}} = \operatorname{mom}(Z - Y, Z - Y) - \mathrm{E}(Z - Y)^2 = \|Z - Y\|^2.$$

It follows that the BLUP is both the projection and the cov-orthogonal projection of Z onto U. Denote $Z^\ddagger := \pi^{\mathrm{cov}}_{\operatorname{span}(X)}(Z)$. Since projections compose, $Z^\dagger_{[X]} = \pi^{\mathrm{cov}}_U(Z^\ddagger)$. Next, denote $W := \{Y \in \operatorname{span}(X) \mid f(Y) = 0, \text{ for all } f \in F\}$. Then $U = W + A$ for every $A \in U$; and for any $Y \in \operatorname{span}(X)$, we have $\pi^{\mathrm{cov}}_U(Y) = \pi^{\mathrm{cov}}_W(Y - A) + A$. Therefore,

$$\begin{aligned} Z^\dagger_{[X]} &= (\mathrm{Id} - \pi^{\mathrm{cov}}_M)(Z^\ddagger - A) + A \\ &= Z^\ddagger - (\langle Z^\ddagger - A, E_q\rangle_{\mathrm{cov}})^{\mathrm{t}}_{q\leq m} G_E^{-1} E \\ &= Z^\ddagger - \sigma^{\mathrm{t}} K^{-1}\Phi^{\mathrm{t}} G_E^{-1} E + \varphi^{\mathrm{t}} G_E^{-1} E. \end{aligned}$$

where $G_E := (\langle E_p, E_q\rangle_{\mathrm{cov}})_{p\leq m, q\leq m}$. The result follows from the facts $E = \Phi K^{-1} X$ and $G_E = \Phi K^{-1}\Phi^{\mathrm{t}}$, and Lemma 2. By Pythagoras' theorem we have $\operatorname{MSE}(Z^\dagger_{[X]}) = \|Z\|^2_{\mathrm{cov}} - \|Z^\ddagger\|^2_{\mathrm{cov}} + \|Z^\ddagger - Z^\dagger_{[X]}\|^2_{\mathrm{cov}}$. By Lemma 3, we have $\|Z^\ddagger\|^2_{\mathrm{cov}} = \sigma^{\mathrm{t}} K^{-1}\sigma$ and $\|Z^\ddagger - Z^\dagger_{[X]}\|^2_{\mathrm{cov}} = (\varphi - \Phi K^{-1}\sigma)^{\mathrm{t}} G_E^{-1}(\varphi - \Phi K^{-1}\sigma)$. □

2.3 Prediction and estimation

Whereas a predictor is a random variable that stands in for another random variable, an estimator is a random variable that stands in for a constant. But any constant may be viewed as a trivial (i.e. constant) random variable. Estimation is therefore a special case of prediction. We are often interested in finding an estimator for the expected value of a random variable. Consider a random variable Z. We may always write this as the sum of its expected value and another, centred random variable, i.e. we may always write $Z = \Theta + A$, where $\Theta := \mathrm{E}(Z)$, and $A := Z - \mathrm{E}(Z)$. We call the BLUP of Θ based on X the *best linear unbiased estimator (BLUE) of* Θ *based on* X. By the BLUP formula (Prop. 6) we find that the BLUE is

$$\Theta^{\dagger}_{[X]} = \varphi^{\mathrm{t}}(\Phi^{\mathrm{t}}K^{-1}\Phi)^{-1}\Phi K^{-1}X$$

with $\mathrm{MSE}(\Theta^{\dagger}_{[X]}) = \varphi^{\mathrm{t}}(\Phi^{\mathrm{t}}K^{-1}\Phi)^{-1}\varphi$. We recognize $\Theta^{\dagger}_{[X]}$ from the Gauss–Markov theorem as the generalized least-squares estimator of Θ based on X.

Remark 7. Following Goldberger (1962) we may rewrite the BLUP formula as

$$Z^{\dagger}_{[X]} = \Theta^{\dagger}_{[X]} + \sigma^{\mathrm{t}}K^{-1}D,$$

where $D := X - \Phi^{\mathrm{t}}(\Phi^{\mathrm{t}}K^{-1}\Phi)^{-1}\Phi K^{-1}X$. We recognize the i-th element of D as $D_i = X_i - \Theta_i{}^{\dagger}_{[X]}$, namely the residual of the BLUE of the expected value of X_i based on X. In this way we see that the BLUP of Z is the sum of the BLUE of the expected value of Z based on X and a weighted sum of the residuals of the best linear unbiased estimators of the the expected values of $X_1, \ldots, X_n$ based on X.

2.4 Prediction intervals

Having found a predictor we might in turn want to find a prediction interval for it. Failing that, we might want to find bounds for such a prediction interval.

2.4.1 A prediction interval

Let Y be a predictor for Z. Suppose that we have a model for the distribution of $Z - (Y - \mathrm{bias}(Y))$ and, furthermore, that we have a pair of γ critical values for this model, c_1 and

c_2, such that

$$c_1\sqrt{\mathrm{var}(Z-(Y-\mathrm{bias}(Y)))} \leq Z-(Y-\mathrm{bias}(Y)) \leq c_2\sqrt{\mathrm{var}(Z-(Y-\mathrm{bias}(Y)))} \quad (3)$$

with probability γ. Note that $\mathrm{var}(Z-(Y-\mathrm{bias}(Y))) = \mathrm{var}(Y-Z)$ since $\mathrm{bias}(Y)$ is a real number. Hence

$$[Y-\mathrm{bias}(Y)+c_1\sqrt{\mathrm{var}(Z-Y)}, Y-\mathrm{bias}(Y)+c_2\sqrt{\mathrm{var}(Z-Y)}] \quad (4)$$

is a γ prediction interval for Z. The bias-variance decomposition gives $\mathrm{MSE}(Y) = \mathrm{var}(Z-Y)+(\mathrm{bias}(Y))^2$. Of course if Y is the BLUP of Z then $\mathrm{bias}(Y)=0$.

2.4.2 Bounds for a prediction interval

Suppose that we do not have a model for the distribution of $Z-Y$, still less a pair of γ critical values for such a model. We may use the empirical rule in the form of Chebyshev's inequality or the Vysochanskij–Petunin inequality to construct bounds for a prediction interval. Then

$$\lambda\sqrt{\mathrm{var}(Z-Y)} \leq Z-(Y-\mathrm{bias}(Y)) \leq \lambda\sqrt{\mathrm{var}(Z-Y)} \quad (5)$$

where $\lambda = 1/\sqrt{1-\gamma}$ or, for the case of unimodal $Z-Y$, $\lambda = 2/(3\sqrt{1-\gamma})$ so long as $Z \neq Y$.

Remark 8 (confidence intervals). Let Y be an estimator for Θ, the mean of Z. We call a prediction interval for Y a 'confidence interval'. A special case of equation 4 is the well-known γ confidence interval for the BLUE of the mean of Z:

$$[\Theta^\dagger_{[X]} + c_1\sqrt{\varphi^{\mathrm{t}}(\Phi^{\mathrm{t}}K^{-1}\Phi)^{-1}\varphi}, \Theta^\dagger_{[X]} + c_2\sqrt{\varphi^{\mathrm{t}}(\Phi^{\mathrm{t}}K^{-1}\Phi)^{-1}\varphi}].$$

2.5 Knowns and unknowns

As it stands the BLP (resp. BLUP) formula gives an abstract relationship between random variables, namely $Z^*_{[X]}$ and X (resp. $Z^\dagger_{[X]}$ and X). The usefulness of this relationship lies in our ability to construct a predictor for Z and its realizations. Given a random vector, X, we would like to know $Z^*_{[X]}$ (resp. $Z^\dagger_{[X]}$) and given a realization of X, namely x, we would

like to know the corresponding realization of $Z^*_{[X]}$ (resp. $Z^\dagger_{[X]}$), namely $z^*_{[X]}$ (resp. $z^\dagger_{[X]}$). In the first case, the BLP (resp. BLUP) is a function $V^n \longrightarrow V$, mapping $X \longmapsto Z^*_{[X]}$, and we think of Z as fixed. In the second case the BLP (resp. BLUP) is a function $\mathbb{R}^n \longrightarrow \mathbb{R}$, mapping $x \longmapsto z^\dagger_{[X]}$. We say a function is *known* if we have an algorithm for computing, to arbitrary precision, the image of an arbitrary element of its domain. Otherwise we say that a function is *unknown*. The BLP (resp. BLUP) formula is a linear combinations of random variables X_i with coefficients formed from various moments of X. In order to know $Z^*_{[X]}$ (resp. $Z^\dagger_{[X]}$) or $z^*_{[X]}$ (resp. $z^\dagger_{[X]}$) we must therefore (i) know how to identity elements of V^n or $\mathbb{R}^n$ and (ii) know the coefficients in the BLP formula (resp. BLUP formula).

We imagine the BLP formula (BLUP formula) being used for either computing the distribution of $Z^*_{[X]}$ (resp. $Z^\dagger_{[X]}$), or for computing a realization $z^*_{[X]}$ (resp. $z^\dagger_{[X]}$). For the first task we identify an element $X \in V^n$ by its joint distribution. The joint distribution itself allows us to compute the coefficients by evaluating the appropriate integrals, and in turn the PDF of $Z^*_{[X]}$ (resp. $Z^\dagger_{[X]}$) can be computed in the normal way, as the distribution of a linear combination of random variables. For the second task we identify an element of $\mathbb{R}^n$ by a real n-tuple. To compute the coefficients we do not need to know the joint distribution of X. Instead of identifying X by its joint distribution we identify it by indexing its elements $X = (X_i)_i$. The random vector $X = (X_i)_{i \leq n}$ is indexed by the set $\{1, \ldots, n\}$. Extending this indexing set to $T := \{1, \ldots, n, n+1\}$, and writing $X_{n+1} := Z$, we may form the random vector $X' = (X_i)_{i \leq n+1}$. We consider the mean-value function $m : T \longrightarrow \mathbb{R}$ given by $m(i) := \mathrm{E}(X_i)$, the second-moment kernel $r : T \times T \longrightarrow \mathbb{R}$ given by $r(i,j) := \mathrm{mom}(X_i, X_j)$, and covariance kernel $k : T \times T \longrightarrow \mathbb{R}$ given by $k(i,j) := \mathrm{cov}(X_i, X_j)$. We imagine the following three states of knowledge.

(K1) We know the second-moment kernel, r, but not the mean-value function, m, or covariance kernel, k. In this situation the BLP and the MSE of the BLP are known, but the bias of the BLP, the BLUP and the MSE of the BLUP are not known.

(K2) We know the covariance kernel, k, but not the mean-value function, m, or second-moment function, r. In this situation the BLUP and MSE of the BLUP are known, but the BLP, the MSE of the BLP, and the bias of the BLP are not known.

(K3) We know the mean-value function, m, and either (equivalently, both) the second-moment kernel, r, or covariance kernel, k. In this situation the BLP, the MSE of the BLP, and the bias of the BLP, along with the BLUP and the MSE of the BLUP, are all known.

Remark 9. In case K3 we may centre Z and X by subtracting $\mathrm{E}(Z)$ and $\mathrm{E}(X)$ respectively. Furthermore, we may choose the very simplest space of pseudoexpectation functions, F, namely the space generated by the single element E. The second and third terms in the BLUP formula (eq. 1) then vanish, as does the third term in its MSE formula (eq. 2). For centred random variables, the second-moment kernel and covariance kernel are identical, and hence we have that $\sigma = \rho$ and $K = R$. Therefore, in this case, the BLP and the BLUP coincide.

Remark 10. It is well known that the best predictor of a random variable Z based on X is the conditional expectation $\mathrm{E}(Z \mid X = x)$, and that if Z and X are centred with joint Gaussian distribution then the best predictor of Z based on X is the BLP of Z based on X. By forming the expectation of a joint Gaussian random vector in this way we may heuristically derive the BLP or the BLUP under the assumption that our state of knowledge is described by case K3 (Rem. 9). This is the derivation used by Rasmussen & Williams (2006). It provides a useful shortcut but it somewhat obscures the difference between the BLP and BLUP as well as the fact that the BLP and BLUP formulas hold for second-order random variables in general.

Remark 11. It is remarkable that we can know the BLP and BLUP, which by definition minimize the MSE, $\mathrm{E}((Y - Z)^2)$ for Y in $\mathrm{span}(X)$ and U respectively, even when we do not know the expectation, E, as in cases K1 and K2. It is remarkable furthermore that having found the BLP and BLUP, we may in turn find their mean-square errors.

3 Curve fitting

Once we have a method for constructing a predictor we may use it to predict the elements of a random process, $Z := (Z_i)_i$. We will call a family of predictors for the elements of Z a 'replicator', and the procedure of constructing replicators we will call 'replication'.

As we have noted, an important use of replication is curve fitting and in particular the fitting of curves to the outputs of expensive computer simulations (Sacks et al. 1989).[3] We adopt the conceit that a curve, z, is the realization of a random process $Z = (Z_t)_{t\in T}$, indexed by the domain of z which is denoted $T \subseteq \mathbb{R}^d$ for some d. Let $(Z_{t_i})_{i\leq n}$ be a finite sample of Z, let $(H_i)_{i\leq n}$ be a centred random vector, and consider the random vector $X := (Z_{t_i} + H_i)_{i\leq n}$, which we view as a sample of Z contaminated by noise. We call a replicator for Z based on X a 'smoother' (also 'filter'). In the case that $H_i = 0$ for all i we call such a smoother an 'interpolator'. If the elements of a smoother (resp. interpolator) are all linear predictors we call it a 'linear smoother based on X' (resp. 'linear interpolator based on X'). Given a realization of X, namely a data set $x = (x_i)_{i\leq n}$, we may compute the equivalent realization of the smoother or interpolator. In fitting a curve to the output of computer simulations, which have no errors, we are interested in constructing interpolators. It is very common to construct such an interpolator using the BLP or BLUP: $Z^*_{[X]} := (Z^*_{t[X]})_{t\in T}$, or $Z^\dagger_{[X]} := (Z^\dagger_{t[X]})_{t\in T}$.

It is trivially true that an arbitrary curve is the realization of *some* random process. But this observation in itself is vacuous. We do not know *which* random process, and hence do not find ourselves in any of the regimes we have considered (Sec. 2.5). We do not know the distribution of this random process. In particular, we do not know its mean, second-moments or covariance. We must therefore choose the mean-value function, second-moment kernel and covariance kernel arbitrarily. This arbitrariness should be alarming, and alert us to the fact that we must test the performance of our linear smoother.

Remark 12. In using the BLP or BLUP as a linear smoother we are not doing regression. In regression we seek an estimator for the mean of a random variable. In the case of the BLP we need know nothing about the mean (case K1), and in the case of the BLUP we perform a regression (by finding the BLUE) to which we add a weighted sum of the residuals (case K2 and Rem. 7).

[3] Although it is an abuse of language, by 'curve' we mean the graph of a function $\mathbb{R}^d \longrightarrow \mathbb{R}$, perhaps only partially defined.

3.1 Validation

The process of testing the performance of a linear smoother is known as 'validation'. Two common types of validation are *cross-validation* and *leave-p-out cross-validation*. Cross-validation is a method for validating predictors based on a fixed X. Alongside X we take another sample of Z, namely $X' = (Z_j)_{j \leq p}$, which we call the 'validation set', and construct a predictor for each $Z_j \in X'$ based on X, namely $Y_{j[X]}$. We define the *cross-validation (CV) predictor residual* of $Y_{j[X]}$ to be

$$d(Y_{j[X]}) = Z_j - Y_{j[X]},$$

and, for a tuple $Y_{[X]} = (Y_{j[X]})_{j \leq p}$ of predictors, we define $d(Y_{[X]}) := (d(Y_{j[X]}))_{j \leq p}$. If the predictors, $Y_{j[X]}$, are estimators for the means of each Z_j then we define the *cross-validation (CV) estimator residual of* $Y_{j[X]}$ to be

$$e(Y_{j[X]}) = Z_j - Y_{j[X]},$$

and, for a tuple $Y_{[X]} = (Y_{j[X]})_{j \leq p}$ of estimators, we define $e(Y_{[X]}) := (e(Y_{j[X]}))_{j \leq p}$. If the distribution of the CV predictor (resp. estimator) residuals are known then we may construct prediction (resp. confidence) intervals for them, otherwise we may construct bounds for their prediction (resp. confidence) intervals (Sec. 2.4). The behaviour of the observed predictor (resp. estimator) residuals should be consistent with the assumptions we have made about their distribution (equivalently, the assumptions we have about the distribution of Z).

In *leave-p-out cross-validation*, instead of using a validation set distinct from X, we partition X into a set containing p elements, X_P, and a set containing $n - p$ elements, $X_{\bar{P}}$. We construct a predictor for each $X_i \in X_P$ based on $X_{\bar{P}}$, namely $Y_{i[X_{\bar{P}}]}$. Then we define the *leave-p-out cross-validation predictor residual* to be $d(Y_{P[X_{\bar{P}}]})$, where $Y_{P[X_{\bar{P}}]} = (Y_{i[X_{\bar{P}}]})_{X_i \in X_P}$. Again, the observed predictor residuals should be consistent with our assumptions about their distribution. Dubrule (1983) and Haslett & Hayes (1998) provided a theory of linear prediction residuals. This relates the leave-p-out BLUP residuals, $d(X_P{}^{\dagger}_{[X_{\bar{P}}]})$, to the BLUE residuals, $d(\Theta^{\dagger}_{[X]})$. The theory may be straightforwardly extended to provide expressions for the leave-p-out BLP and BLUE residuals. First, we consider the

case of the BLP. For the sake of convenience we partition R, $S := R^{-1}$, and X as

$$R = \begin{bmatrix} R_{PP} & R_{P\bar{P}} \\ R_{\bar{P}P} & R_{\bar{P}\bar{P}} \end{bmatrix}, S = \begin{bmatrix} S_{PP} & S_{P\bar{P}} \\ S_{\bar{P}P} & S_{\bar{P}\bar{P}} \end{bmatrix}, X = \begin{bmatrix} X_P \\ X_{\bar{P}} \end{bmatrix}.$$

Proposition 13. *The leave-p-out BLP is*

$$X^*_{P[X_{\bar{P}}]} = -(S_{PP})^{-1} S_{P\bar{P}} X_{\bar{P}}$$

with variance $\operatorname{var}(X^*_{P[X_{\bar{P}}]}) = (S_{PP})^{-1} S_{P\bar{P}} (R_{\bar{P}\bar{P}} - \operatorname{E}(X_{\bar{P}}) \operatorname{E}(X_{\bar{P}})^{\mathrm{t}}) S_{\bar{P}P} (S_{PP})^{-1}$. *The leave-p-out BLP residual is*

$$d(X^*_{P[\bar{P}]}) = (S_{PP})^{-1} [SX]_P$$

where $[SX]_P$ *is the vector consisting of the first* p *rows of* SX. *Its variance is* $\operatorname{var}(d(X^*_{P[\bar{P}]})) = (S_{PP})^{-1} - (S_{PP})^{-1} \operatorname{E}([SX]_P) \operatorname{E}([SX]_P)^{\mathrm{t}} (S_{PP})^{-1}$.

Proof. Denote $A := (R_{\bar{P}\bar{P}})^{-1} X_{\bar{P}}$; then both $X^*_{\bar{P}[X_{\bar{P}}]} = R_{\bar{P}\bar{P}} A$ and $X^*_{P[X_{\bar{P}}]} = R_{P\bar{P}} A$. Since $X^*_{\bar{P}[X_{\bar{P}}]} = X_{\bar{P}}$, we have

$$\begin{bmatrix} R_{PP} & R_{P\bar{P}} \\ R_{\bar{P}P} & R_{\bar{P}\bar{P}} \end{bmatrix} \begin{bmatrix} 0 \\ A \end{bmatrix} = \begin{bmatrix} X^*_{P[X_{\bar{P}}]} \\ X_{\bar{P}} \end{bmatrix}. \tag{6}$$

Pre-multiply this equation by S and extract the first p rows to find that $X^*_{P[X_{\bar{P}}]} = -(S_{PP})^{-1} S_{P\bar{P}} X_{\bar{P}}$ with variance $\operatorname{var}(X^*_{P[X_{\bar{P}}]}) = (S_{PP})^{-1} S_{P\bar{P}} (R_{\bar{P}\bar{P}} - \operatorname{E}(X_{\bar{P}}) \operatorname{E}(X_{\bar{P}})^{\mathrm{t}}) S_{\bar{P}P} (S_{PP})^{-1}$. The leave-p-out CV residuals follow immediately. □

Second, we consider the case of the BLUP. Write the BLUP of X based on X as $X^\dagger_{[X]} = \Phi^{\mathrm{t}} B + KC$ where $B := (\Phi K^{-1} \Phi^{\mathrm{t}})^{-1} \Phi K^{-1} X$, $C := K^{-1} D$, and $D := X - (\Theta_{i[X]})_i$, consistent with Remark 7. Also we define the vector β and matrix Q such that $B = \beta X$ and $C = QX$. We partition K, Φ, Q, β, and X as

$$K = \begin{bmatrix} K_{PP} & K_{P\bar{P}} \\ K_{\bar{P}P} & K_{\bar{P}\bar{P}} \end{bmatrix}, \Phi = \begin{bmatrix} \Phi_P & \Phi_{\bar{P}} \end{bmatrix}, Q = \begin{bmatrix} Q_{PP} & Q_{P\bar{P}} \\ Q_{\bar{P}P} & Q_{\bar{P}\bar{P}} \end{bmatrix}, \beta = \begin{bmatrix} \beta_P & \beta_{\bar{P}} \end{bmatrix}, X = \begin{bmatrix} X_P \\ X_{\bar{P}} \end{bmatrix}.$$

Proposition 14. *The leave-p-out BLUP is*

$$X^\dagger_{P[X_{\bar{P}}]} = -(Q_{PP})^{-1} Q_{P\bar{P}} X_{\bar{P}}$$

with variance $\mathrm{var}(X_{P[X_{\bar{P}}]}^{\dagger}) = (Q_{PP})^{-1}Q_{P\bar{P}}K_{\bar{P}\bar{P}}Q_{\bar{P}P}(Q_{PP})^{-1}$. *The leave-p-out BLUP residual is*

$$d(X_{P[\bar{P}]}^{\dagger}) = (Q_{PP})^{-1}[K^{-1}D]_P$$

where $[K^{-1}D]_P$ *is the vector consisting of the first* p *rows of* $K^{-1}D$. *Its variance is* $\mathrm{var}(d(X_{P[\bar{P}]}^{\dagger})) = (Q_{PP})^{-1}$.

Proof. We closely follow Haslett & Hayes (1998). Note that there exist tuples B' and C' (where $X_{\bar{P}[X_{\bar{P}}]}^{\dagger} = (\Phi_{\bar{P}})^{\mathrm{t}}B' + K_{\bar{P}\bar{P}}C'$) such that

$$\begin{bmatrix} K_{PP} & K_{P\bar{P}} & (\Phi_P)^{\mathrm{t}} \\ K_{\bar{P}P} & K_{\bar{P}\bar{P}} & (\Phi_{\bar{P}})^{\mathrm{t}} \\ \Phi_P & \Phi_{\bar{P}} & 0 \end{bmatrix} \begin{bmatrix} 0 \\ C' \\ B' \end{bmatrix} = \begin{bmatrix} X_{P[X_{\bar{P}}]}^{\dagger} \\ X_{\bar{P}} \\ 0 \end{bmatrix}. \tag{7}$$

Pre-multiply this equation by L, where

$$L := \begin{bmatrix} K & \Phi^{\mathrm{t}} \\ \Phi & 0 \end{bmatrix}^{-1} = \begin{bmatrix} Q & \beta^{\mathrm{t}} \\ \beta & -\mathrm{var}(B) \end{bmatrix},$$

and extract the first p rows to find that $X_{P[X_{\bar{P}}]}^{\dagger} = -(Q_{PP})^{-1}Q_{P\bar{P}}X_{\bar{P}}$ with variance $\mathrm{var}(X_{P[X_{\bar{P}}]}^{\dagger}) = (Q_{PP})^{-1}Q_{P\bar{P}}K_{\bar{P}\bar{P}}Q_{\bar{P}P}(Q_{PP})^{-1}$. The leave-$p$-out CV residuals follow immediately. □

Remark 15. We may find the leave-p-out BLUE and leave-p-out BLUE residual using the same approach. Again, pre-multiply equation 7 by L and this time extract the final m rows to find that

$$B' = B - \beta_P(Q_{PP})^{-1}[K^{-1}D]_P.$$

The leave-p-out BLUE residual is then

$$e(\Theta_{P[X_P]}^{\dagger}) = D_P + (\Phi_P)^{\mathrm{t}}\beta_P(Q_{PP})^{-1}[K^{-1}D]_P$$

since $D_P = X_P - (\Phi_P)^{\mathrm{t}}B$. In the case that $K = \sigma^2 I$, where σ^2 is some constant variance, this reduces to the well-known expression for the ordinary least-squares leave-p-out estimation residuals: $e(\Theta_{P[X_{\bar{P}}]}^{\dagger}) = (I - H_{PP})^{-1}D_P$, where $H := \Phi^{\mathrm{t}}(\Phi^{\mathrm{t}}\Phi)^{-1}\Phi$.

Remark 16. It is common to assume (Rasmussen & Williams 2006) that the fitted curve is the realization of a centred Gaussian random process with second-moment kernel (equivalently, covariance kernel) belonging to one of a standard family of kernels (for example, the Matérn family). In many practical situations this results in predictors that pass validation. The assumption of Gaussianity allows us to compute confidence intervals for the elements of the replicator, but is hard to validate for small sample sizes.

References

Bourbaki, N. (1981), *Espaces vectoriels topologiques*, Springer.

Dubrule, O. (1983), 'Cross validation of kriging in a unique neighborhood', *Journal of the International Association for Mathematical Geology* **15**(6), 687–699.

Goldberger, A. S. (1962), 'Best linear unbiased prediction in the generalized linear regression model', *Journal of the American Statistical Association* **57**(298), 369–375.

Haslett, J. & Hayes, K. (1998), 'Residuals for the linear model with general covariance structure', *Journal of the Royal Statistical Society: Series B (Methodological)* **60**(1), 201–215.

Henderson, C. R. (1963), Selection index and expected genetic advance, *in* W. Hanson & H. Robinson, eds, 'Statistical genetics and plant breeding', The National Academies Press, Washington, DC, pp. 141–163.

Lang, S. (2002), *Algebra*, Springer.

Parzen, E. (1959), Statistical inference on time series by Hilbert space methods, I, Technical report, Stanford University, Stanford.

Rasmussen, C. & Williams, C. (2006), *Gaussian processes for machine learning*, MIT, Cambridge, Mass.

Sacks, J., Welch, W. J., Mitchell, T. J. & Wynn, H. P. (1989), 'Design and analysis of computer experiments', *Statistical Science* **4**(4), 409–423.

Wahba, G. (2003), 'An introduction to reproducing kernel Hilbert spaces and why they are so useful', *IFAC Proceedings Volumes* **36**(16), 525–528. 13th IFAC Symposium on System Identification (SYSID 2003), Rotterdam, The Netherlands, 27-29 August, 2003.

Word count: 3745.