Electronic Journal of Statistics
Vol. 15 (2021) 5310–5332
ISSN: 1935-7524
https://doi.org/10.1214/21-EJS1884

Tuning parameter calibration for personalized prediction in medicine*

Shih-Ting Huang, Yannick Düren, Kristoffer H. Hellton, and Johannes Lederer

Ruhr-University Bochum
Universitätsstraße 150
44801 Bochum
Germany
e-mail: shih-ting.huang@rub.de*;* yannick.dueren@rub.de*;* johannes.lederer@rub.de*; url:* www.johanneslederer.com

Norwegian Computing Center
P.O. Box 114 Blindern
0314 Oslo
Norway
e-mail: hellton@nr.no
url: www.nr.no/~hellton

Abstract: Personalized prediction is of high interest in medicine; potential applications include the prediction of individual drug responses or risks of complications. But typical statistical pipelines such as ridge estimation combined with cross-validation ignore the heterogeneity among the patients and, therefore, are not suited for personalized prediction. We, therefore, introduce an alternative ridge-type pipeline that can minimize the prediction error of each patient individually. We show that our pipeline is optimal in terms of oracle inequalities, fast, and highly effective both in simulations and on real data.

Keywords and phrases: Euclidean distance ridge, high-dimensional estimation, personalized medicine, regularization, ridge regression, tuning parameter calibration.

Received April 2021.

1. Introduction

In the last decade, improvements in genomic, transcriptomic, and proteomic technologies have enabled personalized medicine, or precision medicine, to become an essential part of contemporary medicine. Personalized medicine takes into account individual variability in genes, proteins, environment, and lifestyle to decide on disease prevention and treatment (Hamburg and Collins 2010). The use of a patient's genetic and epigenetic information has already proven to be highly effective to tailor preventive care or drug therapies in a number of applications, such as breast cancer (Cho, Jeon and Kim 2012), prostate cancer (Nam et al. 2007), ovarian cancer (Hippisley-Cox and Coupland 2015), and

*JL and YD acknowledge partial financial support from the DFG project number 451920280.

pancreatic cancer (Ogino et al. 2011), cardiovascular disease (Ehret et al. 2011), cystic fibrosis (Waters et al. 2018), and psychiatry (Demkow and Wolańczyk 2017). The subfield of pharmacogenomics studies specifically how genes affect a person's response to particular drugs to develop more efficient and safer medications (Ziegler et al. 2012). Following Kosorok and Laber (2019) precision medicine may be formalized as sequence of decision rules, a treatment regime, mapping patient information to a recommended action among several different treatments or preventive care.

Genomic, epigenomic, and transcriptomic data used in personalized medicine, such as gene expression, copy number variants, or methylation levels are often high-dimensional with a number of variables that rivals or exceeds the number of observations. Using such data to estimate and predict treatment response or risk of complications, therefore requires regularization typically by the ℓ_1 norm (lasso), the ℓ_2 norm (ridge), or other terms. While there exists tuning-free regularization, such as the methods proposed in (Lederer and Müller 2015; Huang, Xie and Lederer 2021), such methods are not tailored for minimizing the personalized prediction error. On the other hand, regularization often introduces one or more tuning parameters, and these tuning parameters are usually calibrated based on the averaged prediction risks. Most commonly used, K-fold cross-validation (CV) divides the data into K folds (typically $K \in \{5, 10\}$), predicts each fold out-of-sample, averages over all folds for a range of tuning parameters, and selects the value with the lowest averaged error (Stone 1974; Golub, Heath and Wahba 1979). But the averaging removes the inherent individual heterogeneity of the patients and, therefore, results in sub-optimal prediction performance for the individual patients. This may ultimately lead to unsuitable treatment, administration of improper medication with adverse side effects, or lack of preventive care (Hamburg and Collins 2010).

Hence, rather than minimizing an averaged prediction error, our goal is to minimize each patient's individual ("personalized") prediction error. The idea was first introduced by Hellton and Hjort (2018) as a personalized procedure for ridge regression, but they lacked an appropriate calibration scheme for the tuning parameters, utilizing only a naïve plug-in approach, and could therefore not demonstrate a superior predictive performance. In this paper, we introduce an alternative ridge estimator, referred to as Euclidean distance ridge (edr) and calibrate the tuning parameter based on the ideas of adaptive validation (Chichignoud, Lederer and Wainwright 2016) for each patient *individually*. We show that this approach offers compelling theory, fast computations, and accurate prediction on data.

One goal within personalized medicine is to select among multiple treatments (Jeng, Lu and Peng 2018). Our goal, however, is different: we want to predict the risk for a complication or effect of a single treatment as precisely as possible. The specific motivation for our method is to unravel the relationship between gene expression and weight gain in kidney transplant recipients (Cashion et al. 2013). Kidney transplant recipients are known to often gain substantial weight during the first year after transplantation, which can result in adverse health effects (Patel 1998). Individual predictions of this weight gain based on the genetic data can help in identifying high-risk patients.

The standard regularizers are currently sparsity-inducing regularizers such as the ℓ_1 norm (Hastie, Tibshirani and Wainwright 2015). Sparsity is invoked to facilitate interpretation and because applications might be inherently sparse in the first place. However, since we focus sharply on prediction, interpretability is not the key issue, and there is also little evidence for inherent sparsity: quite in contrast, Cashion et al. (2013) already indicates that there might be many genes associated with weight gain. We, therefore, opt for the more classical ridge regression. Ridge regression (Hoerl and Kennard 1970) yields good predictive performance for dense or non-sparse effects, that is, for outcomes related to systemic conditions, as the method does not perform variable selection. Ridge regression is a classical tool for prediction based on genomic data, and it has been shown that ridge regression can outmatch competing prediction methods for survival based on gene expression (Bøvelstad et al. 2007).

From a practical perspective, a short summary of our pipeline is as follows:

1. Compute the ridge estimator for a range of tuning parameters;
2. Translate these estimators into what we call Euclidean distance ridge estimators;
3. Find an optimal tuning parameter for these estimators through a testing scheme.

Our three key contributions of this paper are:

- We introduce a prediction pipeline that takes the heterogeneity among patients into account;
- We develop theoretical guarantees for this pipeline;
- We establish a fast and ready-to-implement algorithm with publicly available code.

The remainder of this paper is organized as follows: We introduce the linear regression framework and the problem statement in Section 2. We then introduce the main methodology of our approach, and present theoretical guarantees in Section 3. In addition, we discuss the algorithm and analyze its performance through simulation studies using synthetic and real data in Section 4. We further apply our pipeline to kidney transplant data in Section 5. Finally, we discuss the results in Section 6 and we defer all proofs to the Appendix A. All data are publicly available and our code is available at https://github.com/LedererLab/personalized_medicine.

2. Problem Setup

We consider data $(\boldsymbol{y}, \boldsymbol{X})$ that follows a linear regression model

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}^* + \boldsymbol{u}. \tag{2.1}$$

Let p denote the number of parameters, e.g. genes or genetic probes, and n the number of samples or patients, then $\boldsymbol{y} \in \mathbb{R}^n$ is the vector of outcomes, y_i, for example, a person's response to treatment. We let $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ denote the design matrix, where each row $\mathbf{x}_i \in \mathbb{R}^p$, $i \in \{1, \dots n\}$, contains the genome information or other covariates of the corresponding person. Each element β_j^*, $j \in \{1, \dots p\}$, of the regression vector $\boldsymbol{\beta}^* \in \mathbb{R}^p$ models the gene's influence on the person's response. We ensure the uniqueness of $\boldsymbol{\beta}^*$ by assuming that it is a projection onto the linear space generated by the n rows of $\boldsymbol{X}$ (Shao and Deng 2012; Bühlmann 2013). For the random error vector $\boldsymbol{u} \in \mathbb{R}^n$, we make no assumptions on the probability distribution.

Our goal is to estimate the regression vector $\boldsymbol{\beta}^*$ from data $(\boldsymbol{y}, \boldsymbol{X})$, or in terms of our application, predicting a person's treatment response based on that person's specific information. Mathematically, this amounts to estimating $\boldsymbol{z}^\top \boldsymbol{\beta}^*$ in terms of the personalized prediction error

$$\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}})\big|, \tag{2.2}$$

where $\boldsymbol{z} \in \mathbb{R}^p$ is what we call the person's "covariate information," which could include genome information as one example.

Since the data in personalized medicine is often high-dimensional, that is, the number of parameters (genes) p rivals or exceeds the number of samples (patients) n, we consider regularized least-squares estimators of the form

$$\hat{\boldsymbol{\beta}}[r] \in \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \big\{ \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + r \cdot f[\boldsymbol{\beta}] \big\}. \tag{2.3}$$

Here, f denotes a function that takes into account prior information, such as sparsity or smaller regression coefficients, and the tuning parameter $r \geq 0$ balances the least-squares term and the prior term. Regularization can also improve prediction accuracy in low-dimensional cases.

Given an estimator (2.3), the main challenge is to find a good tuning parameter in line with our statistical goal. This means that we want to mimic the tuning parameter

$$r^* := \underset{r \in \mathcal{R}}{\arg\min} \big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}[r])\big|, \tag{2.4}$$

which is the optimal tuning parameter in terms of prediction in a given set of candidate parameters $\mathcal{R} := \{r_1, r_2, \dots, r_m\}$.

The optimal tuning parameter r^* depends on the family of estimators (2.3), the unknown noise $\boldsymbol{u}$, and the patient's genome information $\boldsymbol{z}$. The dependence on $\boldsymbol{z}$ is integral to personalized medicine: different patients can respond very differently to the same treatment. But standard tuning-parameter calibration such as CV schemes do not take this personalization into account but instead attempt to minimize the averaged prediction error $\|\boldsymbol{X}\boldsymbol{\beta}^* - \boldsymbol{X}\hat{\boldsymbol{\beta}}[r]\|_2^2/n$ rather than the personalized prediction error $|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}[r])|$. We, therefore, develop a new prediction pipeline that is tailored to the personalized prediction error and equip our methods with fast algorithms and sharp guarantees.

3. Methodology

In this section, we introduce an alternative version of the ridge estimator (Hoerl and Kennard 1970) along with a calibration scheme tailored to personalized medicine. Two distinct features of the pipeline are its finite-sample bounds and its computational efficiency. Our estimator is called *Euclidean distance ridge* (edr) and is defined as

$$\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r] \;\in\; \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min}\bigl\{\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2+r\|\boldsymbol{\beta}\|_2\bigr\}. \tag{3.1}$$

The edr replaces the ridge estimator's squared ℓ_2 prior term $f_{\mathrm{ridge}}[\boldsymbol{\beta}] \equiv \|\boldsymbol{\beta}\|_2^2$ by its square-root $f_{\mathrm{edr}}[\boldsymbol{\beta}] \equiv \sqrt{f_{\mathrm{ridge}}[\boldsymbol{\beta}]} \equiv \|\boldsymbol{\beta}\|_2$. This modification allows us to derive finite-sample oracle inequalities that can be leveraged for tuning-parameter calibration. At the same time, the edr preserves two of the ridge estimator's most attractive features: it can model the influences of many parameters, and it can be computed without the need for elaborate descent algorithms. Finally, we will exploit the theory and method that is developed for the edr to calibrate personalized tuning parameters for ridge regression, see Section 4.

Our first step is to establish finite-sample guarantees for the edr. The key idea is that if the tuning parameter is large enough, the personalized prediction error (2.2) is bounded by a multiple of the tuning parameter. In the main text, we assume an orthonormal design $\boldsymbol{X}^\top\boldsymbol{X}=\boldsymbol{I}_{p\times p}$ for ease of presentation, but we show in the Appendix that this assumptions is required neither in theory (see Appendix B) nor in practice (Appendix D). We establish the following guarantee for edr:

Lemma 3.1 (Oracle inequality for edr). *If* $r \geq 2|(\boldsymbol{X}\boldsymbol{z})^\top\boldsymbol{u}|/(c[\boldsymbol{z},r]\|\boldsymbol{z}\|_2)$, *where*

$$c[\boldsymbol{z},r] := \frac{\bigl|\boldsymbol{z}^\top\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\bigr|}{\|\boldsymbol{z}\|_2\bigl\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\bigr\|_2} \quad \in [0,1],$$

then it holds for orthonormal design that

$$\bigl|\boldsymbol{z}^\top(\boldsymbol{\beta}^*-\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r])\bigr| \leq c[\boldsymbol{z},r]\cdot\|\boldsymbol{z}\|_2\cdot r.$$

Such guarantees are usually called *oracle inequalities* (Lederer et al. 2019). The given oracle inequality is an ideal starting point for our pipeline, because it gives us a mathematical handle on the quality of tuning parameters: a good tuning parameter should be large enough to meet the stated condition and yet small enough to give a sharp bound. The original ridge estimator, however, lacks such inequalities for personalized prediction.

Our proof techniques, which are based on the optimality conditions of the estimator, also yield a similar bound for the original ridge estimator: if $t \geq |(\boldsymbol{X}\boldsymbol{z})^\top\boldsymbol{u}|/\|\boldsymbol{z}\|_2$, then $|\boldsymbol{z}^\top(\boldsymbol{\beta}^*-\hat{\boldsymbol{\beta}}_{\mathrm{ridge}}[t])| \leq |1+\boldsymbol{z}^\top\hat{\boldsymbol{\beta}}_{\mathrm{ridge}}[t]/\|\boldsymbol{z}\|_2|\cdot\|\boldsymbol{z}\|_2\cdot t$. The following pipeline can then be applied the same way as for the edr. But the crucial advantage of the edr's bound is that its right-hand side is bounded by $\|\boldsymbol{z}\|_2\cdot r$, which ensures that the results do not scale with $\boldsymbol{\beta}^*$.

The factor $c[\boldsymbol{z},r]$ can be interpreted as the absolute value of the correlation between the person's covariate information $\boldsymbol{z}$ and the estimator $\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]$. This factor, and therefore $\boldsymbol{z}$, are included in our calibration scheme below, and our pipeline, hence, optimizes the prediction for particular study subjects.

Lemma 3.1 bounds the personalized prediction error of edr as a function of the tuning parameter r. Given $\boldsymbol{z}$, the best tuning parameter in terms of the bound minimizes $c[\boldsymbol{z}, r] \cdot r$ over all tuning parameters, that satisfy the lower bound

$$r \geq \frac{2\big|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}\big|}{c[z, r]\|\boldsymbol{z}\|_2}.$$

The value at the lower bound, which we call the oracle tuning parameter, can be interpreted as the closest theoretical mimic of the optimal tuning parameter r^* from (2.4):

Definition 3.1 (Oracle tuning parameter for personalized prediction)**.** *Given a new person's covariate information $\boldsymbol{z}$, the oracle tuning parameter for personalized prediction in a candidate set $\mathcal{R}$ is given by*

$$r_o \in \operatorname*{arg\,min}_{r \in \overline{\mathcal{R}}} \big\{c[\boldsymbol{z}, r] \cdot r\big\}, \qquad \textit{where} \qquad \overline{\mathcal{R}} := \left\{r \in \mathcal{R} : r \geq \frac{2\big|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}\big|}{c[z, r]\|\boldsymbol{z}\|_2}\right\}.$$

The oracle tuning parameter r_o is the best approximation of the optimal tuning parameter r^* in view of the mathematical theory expressed by Lemma 3.1. In practice, however, one does not know the target $\boldsymbol{\beta}^*$ nor the noise $\boldsymbol{u}$ (typically not even its distribution), such that neither r^* nor r_o are accessible.

Such lower bounds on the tuning parameters and corresponding notions of oracle tuning parameters are standard in high-dimensional regression—see, for example, Bühlmann and van de Geer (2011, Chapter 6); Hastie, Tibshirani and Wainwright (2015); Zhuang and Lederer (2018, Section 2). Broadly speaking, the lower bounds indicate that the tuning parameters need to be chosen sufficiently large to "overrule the noise." The lower bounds typically include $\boldsymbol{X}^\top \boldsymbol{u}$ rather than $\boldsymbol{u}$ directly as a consequence of using Hölder's inequality in the proofs (see the Appendix for our proofs); in other words, the estimators are affected by $\boldsymbol{X}^\top \boldsymbol{u}$ rather than by $\boldsymbol{u}$ directly. The values of the lower bounds are, therefore, called the *effective noise* (Lederer and Vogt 2020, Section 1). Our lower bounds additionally include the personalized quantities $\boldsymbol{z}$ and $c[\boldsymbol{z}, r]$ simply because we consider personalized prediction. Overall, our lower bounds and the oracle tuning parameters are only slight variations of standard notions in high-dimensional statistics.

Our goal is now to estimate r_o in order to match its prediction accuracy (and, therefore, to attempt to reach the accuracy of r^*) with a completely data-driven scheme. Our proposal is based on pairwise tests along the tuning parameter path:

Definition 3.2 (PAV$_{\text{edr}}$: Personalized adaptive validation for edr)**.** *We select a tuning parameter $\hat{r}$ by*

$$\hat{r} \in \operatorname*{arg\,min}_{r \in \mathcal{R}_A} \big\{c[\boldsymbol{z}, r] \cdot r \cdot \|\boldsymbol{z}\|_2\big\}, \tag{3.2}$$

where the set of admissible tuning parameters is

$$\mathcal{R}_A := \left\{ r \in \mathcal{R} \;\middle|\; \max_{\substack{r',r''\in\mathcal{R}\\ r',r''\geq r}} \Big[\big|\boldsymbol{z}^\top(\hat{\boldsymbol{\beta}}_{\text{edr}}[r'] - \hat{\boldsymbol{\beta}}_{\text{edr}}[r''])\big| - (c[\boldsymbol{z},r']\cdot r' + c[\boldsymbol{z},r'']\cdot r'')\|\boldsymbol{z}\|_2 \;\leq 0 \Big] \right\}.$$

The idea of using pairwise tests for tuning-parameter calibration in high-dimensional statistics has been introduced by Chichignoud, Lederer and Wainwright (2016) under the name *adaptive validation*. A difference here is that the factors $c[\boldsymbol{z}, r]\cdot r$ are not constant but depend both on r and $\boldsymbol{z}$. The dependence on $\boldsymbol{z}$ in particular reflects our focus on *personalized* prediction.

The following result guarantees that the data-driven choice $\hat{r}$ indeed provides—up to a constant factor 3—the same performance as the oracle tuning parameter r_o.

Theorem 3.1 (Optimality for personalized adaptive validation for edr)**.** *Under the conditions of Lemma 3.1, it holds that*

$$\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[\hat{r}])\big| \leq 3\, c[\boldsymbol{z}, r_o]\cdot\|\boldsymbol{z}\|_2\cdot r_o.$$

This result guarantees that our calibration pipeline selects an essentially optimal tuning parameter from any grid $\mathcal{R}$. Our pipeline is the only method for tuning parameter selection in personalized medicine that is equipped with such finite-sample guarantees. It does, moreover, not require any knowledge about the regression vector $\boldsymbol{\beta}^*$ nor the noise $\boldsymbol{u}$.

Our calibration method is fully adaptive to the noise distribution; however, it is instructive to exemplify our main result by considering Gaussian noise (see Appendix A.3 for the detailed derivations):

Example 3.1 (Gaussian noise)**.** *Suppose orthonormal design and Gaussian random noise $\boldsymbol{u} \sim \mathcal{N}_n[0_n, \sigma^2\boldsymbol{I}_{n\times n}/n]$. For any $\delta \in (0,1)$, it holds with probability at least $1-\delta$ that*

$$\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[\hat{r}])\big| \leq 3\sigma\sqrt{\frac{8\log(2/\delta)}{n}}\|\boldsymbol{z}\|_2.$$

The bound provides the usual parametric rate $\sigma/\sqrt{n}$ in the number of samples n; the factor $\|\boldsymbol{z}\|_2$ entails the dependence on the number of parameters p.

4. Algorithm and Numerical Analysis

One of the main features of our pipeline is its efficient implementation. This implementation exploits a fundamental property of our estimator: there is a one-to-one correspondence between the edr and the ridge estimator via the tuning parameters.

4.1. Connections to the ridge estimator

The ridge estimator is the ℓ_2^2-regularized least-squares estimator (Hoerl and Kennard 1970)

$$\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] \in \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min}\big\{\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2 + t\|\boldsymbol{\beta}\|_2^2\big\}, \tag{4.1}$$

where $t > 0$ is a tuning parameter. Its computational efficiency, which is due to its closed-form expression, provides a basis for the computation of our edr estimator. The closed-form of the ridge estimator can be derived from the Karush-Kuhn-Tucker (KKT) conditions as

$$\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] = (\boldsymbol{X}^\top\boldsymbol{X} + t\boldsymbol{I}_{p\times p})^{-1}\boldsymbol{X}^\top\boldsymbol{y}, \tag{4.2}$$

noting that the matrix $(\boldsymbol{X}^\top\boldsymbol{X} + t\boldsymbol{I}_{p\times p})$ is always invertible if $t > 0$.

However, the inversion of the matrix $\boldsymbol{X}^\top\boldsymbol{X} + t\boldsymbol{I}_{p\times p}$ still deserves some thought: first, the matrix might be ill-conditioned, and second, the matrix needs to be computed for a range of tuning parameters rather than only for a single one. A standard approach to these two challenges is a singular value decomposition (svd) of the design matrix $\boldsymbol{X}$.

Lemma 4.1 (Computation of the ridge estimator through singular value decomposition)**.** *Let a singular value decomposition of $\boldsymbol{X}$ be given by $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$, where $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{p\times p}$ are orthonormal matrices, and $\boldsymbol{D} = \operatorname{diag}(d_1, d_2, ..., d_p)$ is an $n\times p$ diagonal matrix of the corresponding* singular values $d_1, d_2, ..., d_p$*. Then, the ridge estimator can be computed as*

$$\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] = \boldsymbol{V}\boldsymbol{D}^\dagger\boldsymbol{U}^\top\boldsymbol{y}, \tag{4.3}$$

where $\boldsymbol{D}^\dagger \in \mathbb{R}^{p\times n}$ is diagonal with $\boldsymbol{D}^\dagger = \operatorname{diag}(d_1/(d_1^2+t), ..., d_p/(d_p^2+t))$.

The singular value decomposition of the design matrix does not depend on the tuning parameter; therefore, the ridge estimators $\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]$ can be readily computed for multiple tuning parameters just by substituting the value of t in $\boldsymbol{D}^\dagger$. The resulting set of ridge (edr) estimators for a set of tuning parameters $\mathcal{T}$ is called the ridge (edr) path for $\mathcal{T}$.

Now, the crucial result is that the ridge estimator and the edr are computational siblings.

Theorem 4.1 (One-to-one mapping between tuning parameters)**.** *The one-to-one mapping $\phi[t] : t \mapsto r$ defined by*

$$r = \phi[t] := \big\|2\boldsymbol{X}^\top(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\text{ridge}}[t])\big\|_2 \tag{4.4}$$

transforms tuning parameters t of the ridge estimator to tuning parameters r of the edr estimator such that $\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] = \hat{\boldsymbol{\beta}}_{\text{edr}}[r]$.

This mapping transforms, in particular, the optimal tuning parameter of the ridge estimator to a corresponding optimal tuning parameter of the edr estimator. It can be viewed as a consequence of the edr penalty being a continuous transformation of the ridge penalty. More generally, this mapping allows us to compute the edr estimator via the ridge estimator—see below.

4.2. Algorithm

The core idea of our proposed algorithm is to exploit the above one-to-one mapping between edr estimator and ridge estimator. This correspondence allows us to efficiently compute edr solution paths via the ridge's explicit formulation and svd.

First, consider a set of ridge tuning parameters $\mathcal{T}$ and its corresponding set of edr tuning parameters given by

$$\mathcal{R}_\phi := \big\{r \in \mathbb{R} : \; r = \phi[t], \, t \in \mathcal{T}\big\}$$

with cardinality $m := |\mathcal{R}_\phi|$. This set contains, in particular, the tuning parameter $\hat{r}$, whose optimality is guaranteed under Theorem 3.1. To compute the tuning parameter $\hat{r}$, given data $\boldsymbol{z}$, we first order the elements $r_1, r_2, \ldots, r_m$ of $\mathcal{R}_\phi$ such that

$$c[\boldsymbol{z}, r_1] \cdot r_1 \le c[\boldsymbol{z}, r_2] \cdot r_2 \le \cdots \le c[\boldsymbol{z}, r_m] \cdot r_m. \tag{4.5}$$

The PAV_{edr} method can then be formulated in terms of the binary random variables

$$\hat{s}_{r_i} := \prod_{j=i}^{m} \mathbb{1}\Big\{\big|\boldsymbol{z}^\top(\hat{\boldsymbol{\beta}}_{\text{edr}}[r_i] - \hat{\boldsymbol{\beta}}_{\text{edr}}[r_j])\big| - \big(c[\boldsymbol{z}, r_i] \cdot r_i + c[\boldsymbol{z}, r_j] \cdot r_j\big)\|\boldsymbol{z}\|_2 \;\le\; 0\Big\}$$

for $i \in \{1, \ldots, m\}$, and an algorithm is as follows:

```
Input: $(r_i)_{i=1,\ldots,m}$, $(\hat{\boldsymbol{\beta}}_{\text{edr}}[r_i])_{i=1,\ldots,m}$, $\boldsymbol{z}$
Result: $\hat{r}$

Set initial index: $i \leftarrow m$
while $\hat{s}_{r_i} \neq 0$ and $i > 1$ do
    Update index: $i \leftarrow i - 1$
end
Set output: $\hat{r} \leftarrow r_i$
```

Algorithm 1: Algorithm for PAV_{edr} of Definition 3.2.

The full pipeline can be summarized by the following four steps:

Step 1: Generate a set $\mathcal{T}$ of tuning parameters for ridge regression.
Step 2: Compute the ridge solution path with respect to $\mathcal{T}$ by using (4.3).
Step 3: Transform the ridge tuning parameters to their edr counterparts $\mathcal{R}_\phi$ using (4.4) and sort the tuning parameters according to (4.5).
Step 4: Use the PAV_{edr} method (Algorithm 1) to compute the tuning parameter $\hat{r}$ and map it back to its ridge counterpart $\hat{t}$.

The algorithm can be readily implemented and is fast: it essentially only requires the computation of one ridge solution path (a single svd). In strong contrast, K-fold CV requires the computation of K ridge solution paths. Consequently, the ridge estimator with PAV_{edr} can be computed approximately K times faster than with K-fold CV, which we will confirm in the simulations. Moreover, CV still requires a tuning parameter, namely, the number of folds K, while $\mathrm{PAV}_{\mathrm{edr}}$ is completely parameter-free. We defer a detailed discussion on the complexity and run time of the algorithm to Appendix C.

4.3. Simulation Study

We evaluate the prediction performance of the $\mathrm{PAV}_{\mathrm{edr}}$ method using (1) fully simulated data with random design and (2) a real data set with a simulated outcome. The results are compared to the ridge estimators defined in (4.1) computed by K-fold CV with $K \in \{5, 10\}$, which is a standard reference method, and the Fridge method for personalized prediction (Hellton and Hjort 2018).

The first setting is solely based on simulated data. The dimensions of the design matrix are $(n, p) \in \{(50, 100), (150, 250), (200, 500)\}$. First, the entries of each row of the design matrix $\boldsymbol{X}$ are sampled i.i.d. from $\mathcal{N}[\mu, 1]$, where the mean itself is sampled according to $\mu \sim \mathcal{N}[0, 100^2]$, and the columns of the design matrix are then normalized to have Euclidean norm equal to one. The entries of the regression vector $\boldsymbol{\beta}^*$ are sample i.i.d. from $\mathcal{N}[0, 10^2]$ and then projected onto the row space of $\boldsymbol{X}$ to ensure identifiability (Shao and Deng 2012; Bühlmann 2013). The entries of the noise vector $\boldsymbol{u}$ are sampled i.i.d. from $\mathcal{N}[0, \sigma^2]$, where $\sigma^2 = 2\,\mathrm{Var}[\boldsymbol{X}\boldsymbol{\beta}^*]$ to ensure a signal-to-noise ratio of 0.5. We sample 100 data testing vectors $\boldsymbol{z}$ i.i.d. from $\mathcal{N}[0, 10^2]$ and generate a set of 300 tuning parameters $\mathcal{T} = \{10^q \mid q = -5 + 10i/299,\ i \in \{0, \ldots, 299\}\}$. The mean personalized prediction error $|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}[r])|$ for all 100 data testing vectors $\boldsymbol{z}$ is compared with (i) Fridge method proposed by Hellton and Hjort (2018), (ii) for $r \in \mathcal{T}$ calibrated by $\mathrm{PAV}_{\mathrm{edr}}$, 5-fold CV, and 10-fold CV, (iii) the oracle tuning parameter defined in Definition 3.1, and (iv) the optimal tuning parameter r^* defined in (2.4). The computation is repeated 100 times and results are averaged. To compare the distributions of the mean personalized prediction error obtained by $\mathrm{PAV}_{\mathrm{edr}}$, 5-fold CV, and 10-fold CV, respectively, we further compute the corresponding standard error (SE) that is defined as the standard deviation of all $N := 100 \times 100$ computed personalized prediction errors divided by $\sqrt{N}$.

We observe that in all considered cases, $\mathrm{PAV}_{\mathrm{edr}}$ improves on CV and Fridge both in terms of accuracy as well as in standard error (Table 1). In particular, $\mathrm{PAV}_{\mathrm{edr}}$ mimics the prediction performance of the oracle tuning parameter r_o defined in Definition 3.1 well. While there is still a reasonable discrepancy between the personalized prediction performance of r_o and r^*, we were able to achieve a strong improvement compared to all other tested tuning parameter calibration methods.

In the second setting, we base our simulation on real data for covariates but simulate the outcome. We use the genomic data from the application in Section 5 where the sample size is $n = 26$. The number of covariates in the design matrix is restricted to the $p = 1936$ gene probe targets identified as potentially influential by Cashion et al. (2013). The regression vector and the noise are generated as in the first simulation setting above. The results were averaged over 100 runs

Table 1

Personalized prediction errors for the first simulation setting, which entirely consists of artificial data. $\mathrm{PAV}_{\mathrm{edr}}$ *outperforms* 5*-fold,* 10*-fold CV, and* Fridge *both in accuracy and standard error.*

(n,p)	Method	Mean error	SE
(50,100)	Optimal tuning r^*	65.96	0.77
	Oracle tuning r_o	565.20	4.67
	$\mathrm{PAV}_{\mathrm{edr}}$	570.45	4.30
	Fridge	751.54	21.14
	5-fold CV	8816.08	256.52
	10-fold CV	7181.56	186.64
(150,250)	Optimal tuning r^*	132.71	1.99
	Oracle tuning r_o	959.60	7.30
	$\mathrm{PAV}_{\mathrm{edr}}$	969.88	7.39
	Fridge	1838.47	78.34
	5-fold CV	8702.49	169.75
	10-fold CV	8366.60	161.77
(200,500)	Optimal tuning r^*	156.81	1.85
	Oracle tuning r_o	1129.41	8.69
	$\mathrm{PAV}_{\mathrm{edr}}$	1145.46	8.74
	Fridge	2414.48	81.89
	5-fold CV	5274.06	127.23
	10-fold CV	6411.85	156.43

Table 2

Personalized prediction errors for the the second simulation setting, which consists of real covariate data and simulated outcomes. $\mathrm{PAV}_{\mathrm{edr}}$ *outperforms* 5*-fold,* 10*-fold CV, and* Fridge *again both in accuracy and standard error.*

Method	Mean error	SE
Optimal tuning r^*	10.24	0.09
Oracle tuning r_o	403.54	3.11
$\mathrm{PAV}_{\mathrm{edr}}$	366.48	2.82
Fridge	431.67	3.75
5-fold cross-validated edr	1216.97	35.04
10-fold cross-validated edr	1407.00	36.12

and are summarized in Table 2. We observe again that $\mathrm{PAV}_{\mathrm{edr}}$ improves on CV and Fridge both in terms of accuracy as well as in standard error.

Finally, we investigate a simulation setting where the design matrix $\boldsymbol{X}$ has mutually correlated coordinates with varying degree of correlation. Again, we observe a large improvement in terms of accuracy as well as in standard error of $\mathrm{PAV}_{\mathrm{edr}}$ compared to CV and Fridge. The details and results are deferred to Appendix D. In summary, the results of our simulation studies demonstrate that $\mathrm{PAV}_{\mathrm{edr}}$ is a contender on data, which confirms and complements our theoretical findings from before.

TABLE 3
Personalized prediction errors for in-sample (left) and leave-one-out (right) prediction for the kidney transplant data. Regardless of in-sample or leave-one-out prediction, $\mathrm{PAV}_{\mathrm{edr}}$ *outperforms* 5*-fold,* 10*-fold CV, and* Fridge *again both in accuracy and standard error.*

(a) In-sample prediction

Method	Mean error	SE
$\mathrm{PAV}_{\mathrm{edr}}$	0.0049	0.0023
Fridge	0.0235	0.0034
5-fold CV	0.0457	0.0106
10-fold CV	0.0399	0.0094

(b) Leave-one-out prediction

Method	Mean error	SE
$\mathrm{PAV}_{\mathrm{edr}}$	0.0622	0.0060
Fridge	0.0650	0.0069
5-fold CV	0.0651	0.0099
10-fold CV	0.0680	0.0089

5. Application

Kidney transplant recipients are known to gain significant weight during the first year after transplantation, with a reported average increase of 12 kg (Patel 1998). Such substantial weight gain over a relatively short time period gives an increased risk for several adverse health effects, such as cardiovascular disease, leads to less favorable graft outcomes and may be detrimental for the overall outcome of the patient. The weight gain has been explained by the use of prescribed steroids which increase the appetite, but steroid-free protocols alone have not reduced the risk of obesity, suggesting alternative causes. Even though weight gain is fundamentally caused by a too high calorie intake relative to the energy expenditure, the heterogeneity in the individual response is substantial. Genetic variation has, therefore, been considered as a contributing factor, and several genes have been linked to obesity and weight gain (Bauer et al. 2009; Cheung et al. 2010).

Cashion et al. (2013) investigated whether genomic data can be used to predict weight gain in kidney transplant recipients. This was done by measuring gene expression in subcutaneous adipose tissue which has an important role in appetite regulation and can easily be obtained from the patients during surgery. The patients' weight was recorded at the time of transplantation and at a 6-months follow-up visit, resulting in a relative weight difference. The adipose tissue samples were collected from 26 transplant patients at the time of surgery, and mRNA levels were measured to obtain the gene expression profiles for 28 869 gene probe targets using Affymetrix Human Gene 1.0 ST arrays. All data is publicly available in the EMBL-EBI ArrayExpress database (`www.ebi.ac.uk/arrayexpress`) under accession number E-GEOD-33070. As excessive weight gain can have severe consequences for the patients, the goal is to predict the future weight increase based on the available gene expression profiles. When a large weight increase is predicted, additional efforts assisting with diet restriction and physiotherapy can be set into effect to better tailor the care of each individual patient.

We compare the performance of our method in predicting weight gain for the kidney transplant patients to the prediction of standard ridge regression calibrated by CV. In detail, we make predictions for each patient both in-sample and out-of-sample, leaving out the observation and using the remaining data to fit the penalized regression model and select the optimal tuning parameter. Since we do not know the true parameter $\boldsymbol{\beta}^*$, we can only examine the performance of our method and CV by comparing their estimation errors, which is defined by

$$\left|y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\right|. \tag{5.1}$$

As described in the previous section, the columns of the design matrix are normalized to have Euclidean norm one. Unlike in Section 4.3, we here take all 28 869 gene probes into consideration.

The averaged results are summarized in Table 3a and Table 3b. We observe that $\mathrm{PAV}_{\mathrm{edr}}$ clearly outperforms 5-fold and 10-fold CV for both in-sample and out-of-sample prediction of the kidney transplant data. For out-of-sample prediction, we observe an improvement of about 9.3% in the estimation error compared to 10-fold CV. These improvements, especially in standard deviation, reinforce the advantages of a personalized approach to tuning-parameter calibration.

By predicting the individual weight gain more precisely, our method contributes in ensuring that each patient may get the best possible care. Even though kidney transplant patients are typically encouraged to adhere to a healthy lifestyle (diet and physical activity), obesity prevention can be a difficult to achieve (Cashion et al. 2013). Thus by identifying high-risk patients as precisely as possible, one can better tailor the necessary lifestyle changes, through additional dietary counselling and physical activity. This may prevent the adverse weight gain documented in the first year after transplantation.

6. Conclusion

We have introduced a pipeline that calibrates ridge regression for personalized prediction. Its distinctive features are the finite sample guarantees (see Theorem 3.1) and the statistical and computational efficiency (see Tables 1, 2, and 4). These features are echoed when predicting the weight gain of kidney transplant patients (see Table 3). Hence, our pipeline can improve personalized prediction and, thereby, further the cause of personalized medicine.

One possible limitation of our procedure, relevant for all personalized prediction methods, is that the training data needs to be available to compute each new prediction. This may give certain constraints in terms of data storage and privacy when implementing the procedure in practice or as a commercial product. Nowadays memory tends to not be a problem, in particular for medical dataset where the number of patients is typically small, ranging from tens to hundreds. The memory needed is thus relatively small in view of current online and offline storage capacities. Further, to ensure proper data privacy and software safety any included data would have to be properly anonymized.

Despite our focus on personalized medicine, we also envision applications in other areas where individual heterogeneity is crucial for predictions. Two examples are item recommendation, predicting the rating of an item or product assigned by a specific user (Guy et al. 2010; Rafailidis et al. 2014), and personalized marketing, delivering individualized product prices or messages to specific costumers (Tang, Liao and Sun 2013). Future work includes to extend the methodology to logistic and generalized linear regression and to explore possibilities regarding ℓ_1 or lasso regularization. Further, the improvements in prediction error are also beneficial when selection between different treatments, and our proposed methodology may be extended in the setting of Jeng, Lu and Peng (2018).

Appendix A: Proofs

A.1. Proof of Lemma 3.1

Proof. Assume $r \geq 2|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}|/(c[\boldsymbol{z},r]\|\boldsymbol{z}\|_2)$ and orthonormal design $\boldsymbol{X}^\top\boldsymbol{X} = \boldsymbol{I}_{p\times p}$. According to the KKT conditions of the edr estimator, we have

$$\begin{aligned} r\frac{\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]}{\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\|_2} &= 2\boldsymbol{X}^\top(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]) \\ &= 2\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\beta}^* + \boldsymbol{u} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]) \\ &= 2\boldsymbol{X}^\top\boldsymbol{X}(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]) + 2\boldsymbol{X}^\top\boldsymbol{u}. \end{aligned}$$

Hence,

$$\boldsymbol{X}^\top\boldsymbol{X}(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]) = -\boldsymbol{X}^\top\boldsymbol{u} + \frac{r}{2}\frac{\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]}{\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\|_2}. \tag{A.1}$$

Let $\boldsymbol{z} \in \mathbb{R}^p$ and multiply $\boldsymbol{z}^\top$ from the left to obtain

$$\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]) = -\boldsymbol{z}^\top\boldsymbol{X}^\top\boldsymbol{u} + \frac{r}{2}\frac{\boldsymbol{z}^\top\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]}{\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\|_2}$$

where we use the assumption of orthonormal design. By taking absolute value on both sides and applying the triangle inequality, we derive the following bound for the personalized prediction error (2.2):

$$\begin{aligned} \left|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r])\right| &\leq \left|\boldsymbol{z}^\top\boldsymbol{X}^\top\boldsymbol{u}\right| + \frac{r}{2}\left|\frac{\boldsymbol{z}^\top\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]}{\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\|_2}\right| \\ &\leq \frac{r}{2}c[\boldsymbol{z},r]\|\boldsymbol{z}\|_2 + \frac{r}{2}\left|\frac{\boldsymbol{z}^\top\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]}{\|\boldsymbol{z}\|_2\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\|_2}\right|\|\boldsymbol{z}\|_2 \\ &= c[\boldsymbol{z},r]\cdot r\cdot\|\boldsymbol{z}\|_2, \end{aligned}$$

since $r \geq 2|(\boldsymbol{X}\boldsymbol{z})^\top\boldsymbol{u}|/(c[\boldsymbol{z},r]\|\boldsymbol{z}\|_2)$ by assumption. Finally, we obtain the bound

$$\left|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r])\right| \leq c[\boldsymbol{z},r]\cdot\|\boldsymbol{z}\|_2\cdot r, \tag{A.2}$$

with

$$c[\boldsymbol{z},r] := \frac{\left|\boldsymbol{z}^\top\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\right|}{\|\boldsymbol{z}\|_2\|\hat{\boldsymbol{\beta}}_{\mathrm{edr}}[r]\|_2}. \qquad \square$$

A.2. Proof of Theorem 3.1

Proof. Let $\boldsymbol{z} \in \mathbb{R}^p$ and suppose that the linear regression model (2.1) is under orthonormal design.

Bound on $c[\boldsymbol{z}, \hat{r}] \cdot \hat{r}$: First, we show that $c[\boldsymbol{z}, \hat{r}] \cdot \hat{r} \leq c[\boldsymbol{z}, r_o] \cdot r_o$. Let

$$c[\boldsymbol{z}, \hat{r}] \cdot \hat{r} \geq c[\boldsymbol{z}, r_o] \cdot r_o,$$

then by definition of $\hat{r}$, there must exist two tuning parameters r', r'' with

$$r' \geq \frac{2\big|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}\big|}{c[\boldsymbol{z}, r']\|\boldsymbol{z}\|_2}, \qquad r'' \geq \frac{2\big|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}\big|}{c[\boldsymbol{z}, r'']\|\boldsymbol{z}\|_2},$$

such that

$$\big|\boldsymbol{z}^\top(\hat{\boldsymbol{\beta}}_{\text{edr}}[r'] - \hat{\boldsymbol{\beta}}_{\text{edr}}[r''])\big| \geq \big(c[\boldsymbol{z}, r'] \cdot r' + c[\boldsymbol{z}, r''] \cdot r''\big) \cdot \|\boldsymbol{z}\|_2.$$

However, by Lemma (3.1), we have

$$\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[r'])\big| \leq c[\boldsymbol{z}, r'] \cdot r' \cdot \|\boldsymbol{z}\|_2$$

and

$$\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[r''])\big| \leq c[\boldsymbol{z}, r''] \cdot r'' \cdot \|\boldsymbol{z}\|_2.$$

Applying the triangle inequality to the above displays and combining the results yields

$$\big|\boldsymbol{z}^\top(\hat{\boldsymbol{\beta}}_{\text{edr}}[r'] - \hat{\boldsymbol{\beta}}_{\text{edr}}[r''])\big| \leq \big(c[\boldsymbol{z}, r'] \cdot r' + c[\boldsymbol{z}, r''] \cdot r''\big) \cdot \|\boldsymbol{z}\|_2,$$

which leads to a contradiction to our assumption. Therefore, we obtain the following bound with respect to r_o:

$$c[\boldsymbol{z}, \hat{r}] \cdot \hat{r} \leq c[\boldsymbol{z}, r_o] \cdot r_o.$$

Bound on the personalized prediction error: Since $c[\boldsymbol{z}, \hat{r}] \cdot \hat{r} \leq c[\boldsymbol{z}, r_o] \cdot r_o$, we have

$$\begin{aligned}\big|\boldsymbol{z}^\top(\hat{\boldsymbol{\beta}}_{\text{edr}}[\hat{r}] - \hat{\boldsymbol{\beta}}_{\text{edr}}[r_o])\big| &\leq \big(c[\boldsymbol{z}, \hat{r}] \cdot \hat{r} + c[\boldsymbol{z}, r_o] \cdot r_o\big) \cdot \|\boldsymbol{z}\|_2 \\ &\leq 2 \cdot c[\boldsymbol{z}, r_o] \cdot r_o \cdot \|\boldsymbol{z}\|_2\end{aligned}$$

Applying the triangle inequality, we ultimately find the bound

$$\begin{aligned}\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[\hat{r}])\big| &= \big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[r_o] + \hat{\boldsymbol{\beta}}_{\text{edr}}[r_o] - \hat{\boldsymbol{\beta}}_{\text{edr}}[\hat{r}])\big| \\ &\leq \big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\text{edr}}[r_o])\big| + \big|\boldsymbol{z}^\top(\hat{\boldsymbol{\beta}}_{\text{edr}}[r_o] - \hat{\boldsymbol{\beta}}_{\text{edr}}[\hat{r}])\big| \\ &\leq 3 \cdot c[\boldsymbol{z}, r_o] \cdot r_o \cdot \|\boldsymbol{z}\|_2.\end{aligned}$$

□

A.3. Proof of Example 3.1

Lemma A.1 (Deviation inequality)**.** *For any standard normal variable* $V \sim \mathcal{N}_1[0,1]$*, we have the following concentration bound*

$$\mathbb{P}[|V| \geq x] \leq 2e^{-x^2/2} \qquad (x > 0).$$

Proof. $\mathbb{P}[V > x] = \mathbb{P}[e^{\lambda V} > e^{\lambda x}]$ for all λ. Now by Markov's inequality,

$$\begin{aligned}\mathbb{P}[e^{\lambda V} > e^{\lambda x}] &\leq \frac{\mathbb{E}[e^{\lambda V}]}{e^{\lambda x}} \\ &= e^{\frac{\lambda^2}{2} - \lambda x}\end{aligned}$$

For $\lambda = x$, we have $\mathbb{P}[V > x] \leq e^{-x^2/2}$. Since the standard normal distribution is symmetric about 0, we obtain the desired result. □

Using this concentration bound, we derive the results of Example 3.1.

Proof. Given a $\boldsymbol{z} \in \mathbb{R}^p$, Gaussian noise $\boldsymbol{u} \sim \mathcal{N}_n[0_n, \sigma^2 \boldsymbol{I}_{n\times n}/n]$ with variance σ^2, and suppose that the linear regression model (2.1) is under orthonormal design. We first show that $\mathbb{P}[2|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}|/(c[\boldsymbol{z}, r]\|\boldsymbol{z}\|_2) \geq r_\delta] \leq \delta$ for

$$r_\delta := \frac{\sigma\|\boldsymbol{X}\boldsymbol{z}\|_2}{(c[\boldsymbol{z}, r]\|\boldsymbol{z}\|_2)}\sqrt{\frac{8\log(2/\delta)}{n}}$$

using the concentration bound, Lemma A.1:

$$\begin{aligned}\mathbb{P}\Big[2\big|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}\big|/(c[\boldsymbol{z}, r]\|\boldsymbol{z}\|_2) \geq r_\delta\Big] &= \mathbb{P}\Bigg[\frac{\big|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}\big|}{\frac{\sigma\|\boldsymbol{X}\boldsymbol{z}\|_2}{c[\boldsymbol{z},r]\|\boldsymbol{z}\|_2}\sqrt{1/n}} \geq \frac{c[\boldsymbol{z}, r]\|\boldsymbol{z}\|_2 r_\delta}{2\sigma\|\boldsymbol{X}\boldsymbol{z}\|_2\sqrt{1/n}}\Bigg] \\ &\leq 2\exp\left[-\left(\frac{\sigma\|\boldsymbol{X}\boldsymbol{z}\|_2\sqrt{\frac{8\log(2/\delta)}{n}}}{2\sigma\|\boldsymbol{X}\boldsymbol{z}\|_2\sqrt{1/n}}\right)^2/2\right] \\ &= 2\exp\big[\log[\delta/2]\big] \\ &= \delta.\end{aligned}$$

Hence, $r_\delta \geq 2|(\boldsymbol{X}\boldsymbol{z})^\top \boldsymbol{u}|/c[\boldsymbol{z}, r_o]\|\boldsymbol{z}\|_2$ holds with at least probability $1-\delta$. By Theorem 3.1, we have with at least probability $1-\delta$:

$$\begin{aligned}\big|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}_{\mathrm{edr}}[\hat{r}])\big| &\leq 3\ c[\boldsymbol{z}, r_o]\, r_o\|\boldsymbol{z}\|_2 && (|c[\boldsymbol{z}, r_o]| \leq 1) \\ &= 3\ c[\boldsymbol{z}, r_o]\frac{\sigma\|\boldsymbol{X}\boldsymbol{z}\|_2}{c[\boldsymbol{z}, r_o]\|\boldsymbol{z}\|_2}\sqrt{\frac{8\log(2/\delta)}{n}}\|\boldsymbol{z}\|_2 \\ &= 3\sigma\sqrt{\frac{8\log(2/\delta)}{n}}\|\boldsymbol{z}\|_2. && (\text{orthon. design})\end{aligned}$$

□

A.4. Proof of Lemma 4.1

Proof. Let $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$ be a singular value decomposition of $\boldsymbol{X}$ as given in Lemma 4.1. Then by algebraic manipulation of Equation (4.2) the ridge estimator can be written as

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] &= \left(\boldsymbol{X}^\top\boldsymbol{X} + t\boldsymbol{I}_{p\times p}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{y} \\ &= \left(\boldsymbol{V}\boldsymbol{D}^T\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top + t\boldsymbol{I}_{p\times p}\right)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^\top\boldsymbol{y} \\ &= \left(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^\top + t\boldsymbol{I}_{p\times p}\right)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^\top\boldsymbol{y} \\ &= \boldsymbol{V}\left(\boldsymbol{D}^2 + t\boldsymbol{I}_{p\times p}\right)^{-1}\boldsymbol{V}^\top\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^\top\boldsymbol{y} \\ &= \boldsymbol{V}\boldsymbol{D}^\dagger\boldsymbol{U}^\top\boldsymbol{y},\end{aligned}$$

where the matrix $\boldsymbol{D}^\dagger$ is defined as

$$\boldsymbol{D}^\dagger = \operatorname{diag}\left(\frac{d_1}{d_1^2 + t}, ..., \frac{d_p}{d_p^2 + t}\right). \qquad \square$$

A.5. Proof of Theorem 4.1

Proof. We consider the KKT-conditions of (3.1) and replace the edr estimator with the ridge estimator to obtain

$$r\frac{\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]}{\left\|\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right\|_2} = 2\boldsymbol{X}^\top\left(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right).$$

By taking the ℓ_2-norm of both sides and with $r > 0$, we obtain

$$r = \left\|2\boldsymbol{X}^\top\left(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right)\right\|_2.$$

Thus, we can transform the ridge tuning parameter t to the edr tuning parameter r with respect to the same estimator.

Moreover, there is a one-to-one relationship between edr and ridge. The ridge estimator in (4.2) implies that

$$\left(\boldsymbol{X}^\top\boldsymbol{X} + t\boldsymbol{I}_{p\times p}\right)\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] = \boldsymbol{X}^\top\boldsymbol{y},$$

and hence

$$t\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] = \boldsymbol{X}^\top\left(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right).$$

Since

$$r = \left\|2\boldsymbol{X}^\top\left(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right)\right\|_2 = 2t\left\|\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right\|_2,$$

we have

$$\frac{r}{2\left\|\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\right\|_2} = t$$

and we finally conclude that $\hat{\boldsymbol{\beta}}_{\text{ridge}}[t] = \hat{\boldsymbol{\beta}}_{\text{edr}}[r]$ when $r/2\|\hat{\boldsymbol{\beta}}_{\text{ridge}}[t]\|_2 = t$. $\square$

Appendix B: Beyond Orthogonality

To avoid digression, we have restricted the theories in the main body of the paper to orthonormal design matrices. However, there are straightforward extensions along established lines in high-dimensional theory. In general, the influence of correlation on regularized estimation has been studied extensively—see, for example, Dalalyan, Hebiri and Lederer (2017) and Hebiri and Lederer (2013) for the lasso case. The most straightforward extension of our theories goes via the ℓ_∞-restricted eigenvalue introduced in Chichignoud, Lederer and Wainwright (2016). This condition allows for design matrices, that satisfy $\|\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\delta}\|_\infty \gtrsim \|\boldsymbol{\delta}\|_\infty$ for certain $\boldsymbol{\delta}$. We omit the details; importantly, our simulations demonstrate that our method provides accurate prediction far beyond orthonormal design.

Appendix C: Run Time Measurement

In this section, we compare the complexity and run time of our $\mathrm{PAV}_{\mathrm{edr}}$ algorithm in theory and practice. The $\mathrm{PAV}_{\mathrm{edr}}$ algorithm can be split into two parts: the computation of the ridge solution path and the patient-wise tuning parameter calibration. The ridge solution path is computed using the results of Lemma 4.1 and essentially requires only the computation of a single svd. This computation is independent of the genome information $\boldsymbol{z}$ of new patients and needs to be performed only once.

Algorithm 1 describes the second part of the $\mathrm{PAV}_{\mathrm{edr}}$ method and is computed for each new patient's genome information. The method mainly requires ordering of the bounds $c[\boldsymbol{z}, r_i] \cdot r_i$ and computation of the binary random variables $\hat{s}_{r_i}$ for all tuning parameters $r_i, i = 1, \ldots, m$. Hence, its complexity scales with the number of tuning parameters m, which is fixed; and we use $m = 300$ in all of our simulation studies and applications. Ordering the bounds $c[\boldsymbol{z}, r_i] \cdot r_i$ can be achieved using standard sorting algorithms with average complexity of $\mathcal{O}(m \log m)$. However, for a pair of tuning parameters r_i, r_j with $i < j$ the role of $c[\boldsymbol{z}, r_i], c[\boldsymbol{z}, r_j]$ is very limited in practice and $r_i < r_j$; hence, these bounds can be expected to be largely presorted. Indeed, we observed that all $c[\boldsymbol{z}, r_i] \cdot r_i$ were fully presorted in all of our simulation studies.

We recorded the computational run time of the tuning parameter calibration using $\mathrm{PAV}_{\mathrm{edr}}$, Fridge (Hellton and Hjort 2018), 5-fold CV, and 10-fold CV for both simulation settings (see Section 4.3). All computations were performed in `R` version 4.0.2, and for CV, we used the implementation provided by the `glmnet` package (Friedman, Hastie and Tibshirani 2010). The results (Table 4) demonstrate that $\mathrm{PAV}_{\mathrm{edr}}$ offers feasible and competitive performance in both simulation settings. While $\mathrm{PAV}_{\mathrm{edr}}$ is much slower than Fridge in the patient-independent computation, its patient-wise run time is faster for high (n, p). It is important to note, that since cross-validation only computes a single tuning parameter that is used for all new patients $\boldsymbol{z}_i$, it does not have a patient-wise run time. Hence, for a high number of new patients, CV may be faster in total — however, still much less accurate (compare Section 4.3).

TABLE 4

Run times in seconds measured for tuning parameter calibration using $\mathrm{PAV_{edr}}$*, Fridge, 5-fold CV, and 10-fold CV. The recorded run times are split into a patient-independent and patient-wise runtime, where the patient-wise run time records the mean run time for a single patient. All run times are averaged over 100 runs. The first simulation setting (top) entirely consists of artificial data. The second simulation setting (bottom) consists of real covariate data (see Section 5) and simulated outcomes.* $\mathrm{PAV_{edr}}$ *outperforms 5-fold and 10-fold CV in terms of the patient-independent run time. Fridge is the overall fastest method. CV does not have a patient-wise run time because it does not perform any patient-wise tuning parameter calibration.*

(a) Fully-simulated data.

(n,p)	Method	Run time (in sec)	
		patient-independent	patient-wise
(50,100)	$\mathrm{PAV_{edr}}$	0.09	0.15
	Fridge	0.01	0.05
	5-fold CV	4.21	-
	10-fold CV	8.09	-
(150,250)	$\mathrm{PAV_{edr}}$	0.65	0.18
	Fridge	0.11	0.27
	5-fold CV	17.50	-
	10-fold CV	33.33	-
(200,500)	$\mathrm{PAV_{edr}}$	1.62	0.23
	Fridge	0.26	0.66
	5-fold CV	27.57	-
	10-fold CV	57.33	-

(b) Real covariate data, simulated outcomes ($n = 26$, $p = 1936$).

Method	Run time (in sec)	
	patient-independent	patient-wise
$\mathrm{PAV_{edr}}$	0.33	0.73
Fridge	0.01	0.27
5-fold CV	2.01	-
10-fold CV	3.71	-

Appendix D: SIMULATION STUDY

In this section, we perform a simulation study for correlated covariates and, hence, apply our method to non-orthogonal design matrices. We sample each row of the design matrix $\boldsymbol{X}$ from a p-dimensional normal distribution $\mathcal{N}[\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p]$. Here, the mean vector $\boldsymbol{\mu}_p \in \mathbb{R}^p$ is defined as $\boldsymbol{\mu}_p := (\mu, \dots, \mu)^\top$ such that μ is sampled from $\mathcal{N}[0, 100^2]$ and the covariance matrix $\boldsymbol{\Sigma}_p \in \mathbb{R}^{p \times p}$ is given by $\boldsymbol{\Sigma}_p := (1-k)\boldsymbol{I} + k\mathbb{1}$, where $\mathbb{1} := (1, \dots, 1)^\top (1, \dots, 1)$ is the matrix of ones and $k \in \{0, 0.2, 0.4\}$ is the magnitude of the mutual correlations. All other settings including the dimensions of $\boldsymbol{X}$, regression vector $\boldsymbol{\beta}^*$, noise vector $\boldsymbol{u}$, signal-to-noise ratio, testing vectors, and tuning parameters are the same as in Section 4.3.

The personalized prediction error $|\boldsymbol{z}^\top(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}[r])|$ is averaged over 100 data vectors $\boldsymbol{z}$ where the tuning parameter $r \in \mathcal{T}$ is calibrated using $\mathrm{PAV_{edr}}$, 5-fold CV, and 10-fold CV. We run 100 experiments for each set of parameters and report the averaged results in Tables 5, 6, and 7 where the run time is scaled rel-

Table 5
Personalized prediction errors for the simulation setting with correlated covariates for $\{n, p\} = \{50, 100\}$. PAV_{edr} *outperforms* 5*-fold,* 10*-fold CV, and* Fridge *in accuracy and standard error.*

n	p	k	Method	Mean error	SE
50	100	0	Optimal tuning r^*	74.54	0.92
50	100	0	Oracle tuning r_o	554.62	4.16
50	100	0	PAV_{edr}	555.75	4.61
50	100	0	Fridge	918.50	51.31
50	100	0	5-fold CV	8734.20	226.55
50	100	0	10-fold CV	9847.69	274.48
50	100	0.2	Optimal tuning r^*	60.12	0.75
50	100	0.2	Oracle tuning r_o	549.92	4.21
50	100	0.2	PAV_{edr}	556.11	4.27
50	100	0.2	Fridge	679.48	15.07
50	100	0.2	5-fold CV	9153.27	238.31
50	100	0.2	10-fold CV	10637.90	293.39
50	100	0.4	Optimal tuning r^*	46.05	0.47
50	100	0.4	Oracle tuning r_o	566.93	4.34
50	100	0.4	PAV_{edr}	571.57	4.39
50	100	0.4	Fridge	633.44	11.61
50	100	0.4	5-fold CV	9036.80	213.18
50	100	0.4	10-fold CV	10810.48	234.74

Table 6
Personalized prediction errors for the simulation setting with correlated covariates for $\{n, p\} = \{150, 250\}$. PAV_{edr} *outperforms* 5*-fold,* 10*-fold CV, and* Fridge *in accuracy and standard error.*

n	p	k	Method	Mean error	SE
150	250	0	Optimal tuning r^*	123.01	1.65
150	250	0	Oracle tuning r_o	981.10	7.42
150	250	0	PAV_{edr}	985.64	7.50
150	250	0	Fridge	1511.52	46.95
150	250	0	5-fold CV	8218.37	196.53
150	250	0	10-fold CV	9747.13	250.32
150	250	0.2	Optimal tuning r^*	112.23	1.15
150	250	0.2	Oracle tuning r_o	974.30	7.45
150	250	0.2	PAV_{edr}	992.19	7.45
150	250	0.2	Fridge	1238.27	34.71
150	250	0.2	5-fold CV	7297.49	167.94
150	250	0.2	10-fold CV	7008.09	158.10
150	250	0.4	Optimal tuning r^*	109.72	1.24
150	250	0.4	Oracle tuning r_o	975.78	7.36
150	250	0.4	PAV_{edr}	983.09	7.41
150	250	0.4	Fridge	1182.14	28.97
150	250	0.4	5-fold CV	9428.55	217.59
150	250	0.4	10-fold CV	7408.99	157.15

TABLE 7

Personalized prediction errors for the simulation setting with correlated covariates for $\{n, p\} = \{200, 500\}$. $\mathrm{PAV}_{\mathrm{edr}}$ *outperforms* 5*-fold,* 10*-fold CV, and* Fridge *in accuracy and standard error.*

n	p	k	Method	Mean error	SE
200	500	0	Optimal tuning r^*	155.80	2.27
200	500	0	Oracle tuning r_o	1123.73	8.59
200	500	0	$\mathrm{PAV}_{\mathrm{edr}}$	1138.62	9.13
200	500	0	Fridge	2106.23	61.65
200	500	0	5-fold CV	9615.19	296.89
200	500	0	10-fold CV	9455.81	269.50
200	500	0.2	Optimal tuning r^*	161.19	2.26
200	500	0.2	Oracle tuning r_o	1125.55	8.56
200	500	0.2	$\mathrm{PAV}_{\mathrm{edr}}$	1137.33	8.67
200	500	0.2	Fridge	2746.95	108.81
200	500	0.2	5-fold CV	10967.50	362.05
200	500	0.2	10-fold CV	12045.71	352.02
200	500	0.4	Optimal tuning r^*	188.15	3.31
200	500	0.4	Oracle tuning r_o	1123.57	8.46
200	500	0.4	$\mathrm{PAV}_{\mathrm{edr}}$	1136.71	8.64
200	500	0.4	Fridge	2031.13	84.694
200	500	0.4	5-fold CV	9432.63	241.50
200	500	0.4	10-fold CV	8181.57	217.79

ative to $\mathrm{PAV}_{\mathrm{edr}}$. We observe that $\mathrm{PAV}_{\mathrm{edr}}$ clearly outperforms 5-fold, 10-fold CV, and Fridge both in terms of accuracy as well as in standard error in all considered cases. This demonstrates that $\mathrm{PAV}_{\mathrm{edr}}$ can effectively account for individual heterogeneity of the data vectors even in cases of correlated covariates.

References

Bauer, F., Elbers, C. C., Adan, R. A., Loos, R. J., Onland-Moret, N. C., Grobbee, D. E., van Vliet-Ostaptchouk, J. V., Wijmenga, C. and van der Schouw, Y. T. (2009). Obesity genes identified in genome-wide association studies are associated with adiposity measures and potentially with nutrient-specific food preference. *Am. J. Clin. Nutr.* **90** 951–959.

Bøvelstad, H. M., Nygård, S., Størvold, H. L., Aldrin, M., Borgan, Ø., Frigessi, A. and Lingjærde, O. C. (2007). Predicting survival from microarray data-a comparative study. *Bioinformatics* **23** 2080–2087.

Bühlmann, P. (2013). Statistical significance in high-dimensional linear models. *Bernoulli* **19** 1212–1242. MR3102549

Bühlmann, P. and van de Geer, S. (2011). *Statistics for high-dimensional data: Methods, theory and applications. Springer Series in Statistics.* MR2807761

Cashion, A., Stanfill, A., Thomas, F., Xu, L., Sutter, T., Eason, J., Ensell, M. and Homayouni, R. (2013). Expression levels of obesity-related genes are associated with weight change in kidney transplant recipients. *PloS One* **8** e59962.

Cheung, C. Y., Tso, A. W., Cheung, B. M., Xu, A., Ong, K., Fong, C. H., Wat, N. M., Janus, E. D., Sham, P. C. and Lam, K. S. (2010). Obesity susceptibility genetic variants identified from recent genome-wide association studies: implications in a chinese population. *J. Clin. Endocrinol. Metab.* **95** 1395–1403.

Chichignoud, M., Lederer, J. and Wainwright, M. J. (2016). A Practical Scheme and Fast Algorithm to Tune the Lasso With Optimality Guarantees. *J. Mach. Learn. Res.* **17** 1–20. MR3595165

Cho, S.-H., Jeon, J. and Kim, S. I. (2012). Personalized medicine in breast cancer: a systematic review. *J. Breast Cancer* **15** 265–272.

Dalalyan, A. S., Hebiri, M. and Lederer, J. (2017). On the prediction performance of the Lasso. *Bernoulli* **23** 552–581. MR3556784

Demkow, U. and Wolańczyk, T. (2017). Genetic tests in major psychiatric disorders–integrating molecular medicine with clinical psychiatry–why is it so difficult? *Transl. Psychiat.* **7** 1–9.

Ehret, G. B., Munroe, P. B., Rice, K. M., Bochud, M., Johnson, A. D., Chasman, D. I., Smith, A. V., Tobin, M. D., Verwoert, G. C., Hwang, S.-J. et al. (2011). Genetic variants in novel pathways influence blood pressure and cardiovascular disease risk. *Nature* **478** 103–109.

Friedman, J., Hastie, T. and Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software* **33** 1–22.

Golub, G. H., Heath, M. and Wahba, G. (1979). Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics* **21** 215–223. MR0533250

Guy, I., Zwerdling, N., Ronen, I., Carmel, D. and Uziel, E. (2010). Social media recommendation based on people and tags. In *Proceedings of the 33rd international ACM SIGIR conference on Research and Development in Information Retrieval* 194–201. ACM.

Hamburg, M. A. and Collins, F. S. (2010). The path to personalized medicine. *New Engl. J. Med.* **363** 301–304.

Hastie, T., Tibshirani, R. and Wainwright, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. Chapman&Hall/CRC. MR3616141

Hebiri, M. and Lederer, J. (2013). How Correlations Influence Lasso Prediction. *IEEE Trans. Inf. Theorie* **59** 1846–1854. MR3030757

Hellton, K. H. and Hjort, N. L. (2018). Fridge: Focused fine-tuning of ridge regression for personalized predictions. *Stat. Med.* **37** 1290–1303. MR3777975

Hippisley-Cox, J. and Coupland, C. (2015). Development and validation of risk prediction algorithms to estimate future risk of common cancers in men and women: prospective cohort study. *BMJ Open* **5** 1–25.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **12** 55–67.

Huang, S.-T., Xie, F. and Lederer, J. (2021). Tuning-free ridge estimators for high-dimensional generalized linear models. *Computational Statistics & Data Analysis* 107205. MR4229301

Jeng, X. J., Lu, W. and Peng, H. (2018). High-dimensional inference for personalized treatment decision. *Electronic journal of statistics* **12** 2074. MR3816967

Kosorok, M. R. and Laber, E. B. (2019). Precision medicine. *Annual review of statistics and its application* **6** 263–286. MR3939521

Lederer, J. and Müller, C. (2015). Don't fall for tuning parameters: tuning-free variable selection in high dimensions with the TREX. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Lederer, J. and Vogt, M. (2020). Estimating the Lasso's Effective Noise. *arXiv:2004.11554*.

Lederer, J., Yu, L., Gaynanova, I. et al. (2019). Oracle inequalities for high-dimensional prediction. *Bernoulli* **25** 1225–1255. MR3920371

Nam, R. K., Toi, A., Klotz, L. H., Trachtenberg, J., Jewett, M. A., Appu, S., Loblaw, D. A., Sugar, L., Narod, S. A. and Kattan, M. W. (2007). Assessing individual risk for prostate cancer. *J. Clin. Oncol.* **25** 3582–3588.

Ogino, S., Galon, J., Fuchs, C. S. and Dranoff, G. (2011). Cancer immunology—analysis of host and tumor factors for personalized medicine. *Nat. Rev. Clin. Oncol.* **8** 711–719.

Patel, M. G. (1998). The effect of dietary intervention on weight gains after renal transplantation. *J. Ren. Nutr.* **8** 137–141.

Rafailidis, D., Axenopoulos, A., Etzold, J., Manolopoulou, S. and Daras, P. (2014). Content-based tag propagation and tensor factorization for personalized item recommendation based on social tagging. *ACM Trans. Interact. Intell. Syst.* **3** 1–26.

Shao, J. and Deng, X. (2012). Estimation in high-dimensional linear models with deterministic design matrices. *Ann. Stat.* **40** 812–831. MR2933667

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *J. R. Statist. Soc. B* **36** 111–133. MR0356377

Tang, H., Liao, S. S. and Sun, S. X. (2013). A prediction framework based on contextual data to support mobile personalized marketing. *Decis. Support Syst.* **56** 234–246.

Waters, S. A. N., Wong, S. L. I., Awatade, N. T., Hewson, C. K., Fawcett, L. K., Kicic, A. and Jaffe, A. (2018). Human primary epithelial cell models: promising tools in the era of cystic fibrosis Personalized Medicine. *Front. Pharmacol.* **9** 1–11.

Zhuang, R. and Lederer, J. (2018). Maximum regularized likelihood estimators: a general prediction theory and applications. *Stat* **7** e186. MR3816901

Ziegler, A., Koch, A., Krockenberger, K. and Grosshennig, A. (2012). Personalized medicine using DNA biomarkers: a review. *Hum. Genet.* **131** 1627–1638.