Qualifying Paper Report for Unsupervised Learning with Stein's Unbiased Risk Estimator

Naitong Chen

October 24, 2022

1 Summary

Parameter estimation lies in the heart of statistical inference, and the customary maximum likelihood estimator (MLE) may not be optimal in terms of the mean-squared error (MSE). Consider the setting where for some $n \in \mathbb{N}$, we have an observation $x \in \mathbb{R}^n$ that is a realization of $X \sim \mathcal{N}(\mu, I)$. To estimate $\mu \in \mathbb{R}^d$, maximum likelihood estimation would yield $\hat{\mu}(x) = x$. In Stein (1956), a perhaps surprising result shows that when $n \geq 3$, there exists some other estimator $\tilde{\mu}$ such that

$$\mathbb{E}\|\tilde{\mu}(X) - \mu\|^2 < \mathbb{E}\|\hat{\mu}(X) - \mu\|^2.$$

In fact, there are many other cases where the maximum likelihood estimator is not optimal under the MSE, a widely used metric for evaluating the quality of an estimator thanks to its mathematical tractability (Berger, 1975; DeGroot, 2005). In a follow-up work by Charles Stein, he developed what is known as Stein's unbiased risk estimate (SURE), which provides an unbiased estimate of the MSE of an arbitrary estimator for the mean of a normally distributed random variable of the form $\mathcal{N}(\mu, \sigma^2 I)$. In what follows, we present a version of this result as outlined in Tibshirani and Wasserman (2015).

Lemma 1.1. *Let $X \sim \mathcal{N}(\mu, \sigma^2 I)$, where $\mu \in \mathbb{R}^n$ and $\sigma > 0$. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a function, and let $f(\cdot, x_{-i})$ refer to f as a function of its i^{th} component x_i with all other components x_{-i} held fixed. Suppose for each $i = 1, \cdots, n$ and almost every $x_{-i} \in \mathbb{R}^{n-1}$, $f(\cdot, x_{-i}) : \mathbb{R} \to \mathbb{R}$ is absolutely continuous. If we further assume $\mathbb{E}\|f(X)\|_2 < \infty$, then*

$$\frac{1}{\sigma^2}\mathbb{E}\left[(X - \mu)f(X)\right] = \mathbb{E}\left[\nabla f(X)\right].$$

By decomposing f by its coordinate functions $f = (f_1, \ldots, f_n)$, we have that for each $i = 1, \ldots, n$,

$$\frac{1}{\sigma^2}\mathbb{E}\left[(X - \mu)f_i(X)\right] = \mathbb{E}\left[\nabla f_i(X)\right].$$

Then summing over all n components yields

$$\frac{1}{\sigma^2}\sum_{i=1}^{n}\mathrm{Cov}(X_i, f_i(X)) = \frac{1}{\sigma^2}\sum_{i=1}^{n}\mathbb{E}\left[(X_i - \mu_i)f_i(X)\right] = \mathbb{E}\left[\sum_{i=1}^{n}\frac{\partial f_i}{\partial X_i}(X)\right].$$

Now suppose $\hat{\mu} : \mathbb{R}^n \to \mathbb{R}^n$ is an arbitrary estimator that satisfies the assumptions laid out in Lemma 1.1, it can be shown that

$$R = \mathbb{E}\|\mu - \hat{\mu}(X)\|^2 = -n\sigma^2 + \mathbb{E}\|X - \hat{\mu}(X)\|^2 + 2\sum_{i=1}^{n}\mathrm{Cov}\left(X_i, \hat{\mu}_i(X)\right),$$

which finally leads to

$$\hat{R} = -n\sigma^2 + \|X - \hat{\mu}(X)\|^2 + 2\sigma^2\sum_{i=1}^{n}\frac{\partial \hat{\mu}_i}{\partial X_i}(X)$$

as an unbiased estimator for the MSE of $\hat{\mu}$.

It is worth noting that the SURE can be employed on a very general class of estimators and that it removes the explicit dependence on the unknown μ. These desirable features have enabled SURE to fuel the development of many estimators that are more superior in MSE than the MLE for parameter estimation problems under the normal distribution and beyond. For instance, under the SURE framework, the James-Stein estimator (James and Stein, 1992) can be shown to be a strictly better estimator in terms of MSE for normally distributed vectors with unit covariance. Lemma 1.1 has also been extended to the exponential family, where subsequent estimators outperforming the MLE in terms of MSE have been developed for parameter estimation problems when the underlying distribution is Gamma, Poisson, ect. (Hudson, 1978; Peng, 1975; Tsui, 1978).

1.1 SURE in model selection

The SURE has also been found in a wide range of applications beyond merely parameter estimation. As an example, it can be used to perform model selection for ridge regression. In a typical linear regression setting, we are given a set of n observations such that

$$y_i = x_i^T \beta + \epsilon_i,$$

where $\beta \in \mathbb{R}^p$ for some $p \in \mathbb{N}$ and for all $i = 1, \ldots, n$, $x_i \in \mathbb{R}^p$, $\epsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$ for some $\sigma > 0$. We can equivalently write that

$$y = \begin{bmatrix} y_1 & \cdots & y_n \end{bmatrix}^T \sim \mathcal{N}\left(X\beta, \sigma^2 I\right), \quad \text{where } X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T.$$

Note that for the purpose of this report, we assume that the data have been centred, and so an intercept term need not be included. Given a regularization parameter $\lambda \geq 0$, we can set

$$\hat{\mu}_\lambda(y) = \hat{\beta}_{\text{ridge},\lambda} = \left(X^T X + \lambda I\right)^{-1} X^T y,$$

the ridge estimator for β, and subsequently the unbiased risk estimate for $\hat{\mu}_\lambda$ takes the form

$$\hat{R}(\lambda) = -n\sigma^2 + \|y - \hat{\mu}_\lambda(y)\|^2 + 2\sigma^2 \sum_{i=1}^{n} \frac{\partial \hat{\mu}_{\lambda,i}}{\partial y_i}(y).$$

Note that the second term encourages the estimates to be close to the observations, and the last term encourages the estimator to not change much under perturbations of the observations, thus creating a bias-variance trade-off. As a result, selecting $\lambda \geq 0$ by minimizing the SURE can be seen as a model selection procedure that is similar in spirit to cross-validation. This λ selection procedure was first proposed in Mallows (1973), and the corresponding risk estimate was later shown in Li (1986) to be asymptotically optimal as the numbser of observations approaches infinity. Namely, denote the selected regularization parameter $\lambda^\star$, we have that

$$\frac{\hat{R}\left(\lambda^\star\right)}{\inf_{\lambda \geq 0} R(\lambda)} \xrightarrow{p} 1.$$

1.2 SURE in image denoising

SURE has also been widely used in the application of image denoising, where it is most commonly used directly as the objective function under which we find the optimal parameter setting using a set of training images (noisy and noise-less). Typically these parameters control the threshold used to decide whether the corresponding signal should be removed. While SURE directly applies when the noise is assumed to be normally distributed, there have also been methods developed to handle other distributions of noise (Donoho and Johnstone, 1995; Luisier et al., 2010; Panisetti et al., 2014). Using a slightly different approach, the SURE framework has also been shown in Metzler et al. (2018) to be particularly useful in the setting where training images are not available or only noisy images are available without their noise-less counterparts. This is indeed a very common setting in practice: in medical imaging, microscopy, and astronomy, noise-less ground truth data are rarely available. Here we discuss this work in more detail. Suppose that for an unobserved noise-less iamge $x \in \mathbb{R}^n$, we observe a noisy version of the image y such that

$$y = x + w, \quad w \sim \mathcal{N}\left(0, \sigma^2 I\right).$$

Our goal is to recover the noise-less image x by transforming the noisy observation y through some image denoiser $f_\theta : \mathbb{R}^n \to \mathbb{R}^n$ parameterized by θ. Under the framework of SURE, we can reconstruct the image using the optimal denoiser function obtained by minimizing the unbiased risk estimate

$$\hat{R}(\theta) = -\sigma^2 + \|y - f_\theta(y)\|^2 + 2\sigma^2 \sum_{i=1}^{n} \frac{\partial f_{\theta,i}}{\partial y_i}(y) \approx R(\theta) = \mathbb{E}_w \|x - f_\theta(y)\|^2.$$

The key observations here is that, under the framework of SURE, we no longer require the noise-less ground truth image to obtain a reconstructed image that minimizes the MSE between itself and the true image. Through the lense of bias-variance trade-off, this above formulation also naturally balances between obtaining an approximation close to the observation and overfitting to the noise in the observed image. Therefore, given any noisy image, we can obtain a reconstructed image that minimizes the MSE under normally distributed noise. Furthermore, if we were given a training set of K noisy observations $(y_k)_{k=1}^K$, we can train a denoiser that generalizes to the class of images contaminated with normally distributed noise by minimizing the sum of these individual risk estimates.

While SURE acts as a natural device for image deblurring in the absense of ground truth noise-less training images, there is one major challenge of this approach. Namely, it is difficult to compute the gradient of f_θ with respect to y, often referred to as the divergence. Many of the modern-day image denoisers are neural networks with extremely complicated structure that makes computing its gradient by hand difficult (Dong et al., 2014; Yang et al., 2017; Zhang et al., 2017). However, to optimize the unbiased risk estimate over θ using automatic differentiation as proposed in Metzler et al. (2018), direct computation or approximation of the gradient of f_θ with respect to y is often required. This is because nested automatic differentiation is not typically supported in existing packages. This issue is addressed by resorting to finite-difference type approximation of the divergence term introduced in MC-SURE (Ramani et al., 2008). However, this method still requires the user to specify the spacing parameter ϵ, and the effect of ϵ on the resulting image denoisers remains unexplored.

2 Mini-proposals

2.1 Proposal 1: Using SURE to automate cross validation for principal component regression with L1 regularization

2.1.1 Motivation and problem formulation

Principal component regression (PCR), a regression method based on principal components (PCs) obtained from principal component analyses (PCA), is commonly used in genomics applications among others. This is because, rather than the effect of individual gene expressions, practitioners are typically more interested in studying the effect of groups of genes, which may describe more complicated processes (Ding et al., 2022).

The general procedure of PCR is typically as follows. Suppose we are given a centred data matrix $X \in \mathbb{R}^{n \times p}$ and corresponding responses $y \in \mathbb{R}^n$ consisting of n observations each with p predictors. Let $X = U\Sigma V^T$ denote the singular value decomposition of the data matrix X, then $W = XV$ denotes the set of all PCs, a set of transformed feature vectors that form an orthonormal basis, ordered by the amount of variance from the data that each PC explains. One can then perform linear regression on the PCs against the response y. Because the PCs are orthogonal to each other, this approach usually leads to better numerical stability than, for example, ordinary least squares.

When the number of predictors is large, a common practice is to perform PCR using only the top k PCs ($k \ll p$), i.e., the PCs that explain the most variance present in the data (Cera et al., 2019; Harel et al., 2019). However, it is important to note that the PCs are fit without knowledge of the response variable, and so the idea that performing PRC using the first k PCs will lead to good a good fit of the data is merely a heuristic. In fact, Jolliffe (1982) provides a number of real examples where performing PCR using the first few PCs that explain the most variance of the data is indeed suboptimal.

As a result, it is desirable to do feature selection among the PCs so that we can balance computational cost and the quality of our PCR model. From the previous section, we know that the SURE can be used as a model selection tool for selecting the regularization parameter for ridge regression. It is then natural to look into extending this framework to LASSO regression on the PCs, which can help us pick the PCs to include in the regression while taking into account the bias-variance trade-off.

2.1.2 Proposed approach

Given the transformed data matrix W, a response vector y consisting of n observations, and a regularization parameter $\lambda \geq 0$, LASSO regression finds the regression coefficients of the form

$$\hat{\beta}_{L,\lambda}(y) = \arg\min_{\beta} \frac{1}{n}\|W\beta - y\|_2^2 + \lambda\|\beta\|_1.$$

This L1 penalty term encourages sparse solutions where the regression coefficient for some the transformed predictors are set to 0. To use the SURE framework to select a subset of the PCs, we require that 1) the resulting estimator $\hat{\beta}_{L,\lambda}$ to be available in closed-form, and 2) we can compute the derivative of this estimator with respect to y.

In the context of PCR, we know that W is an orthonormal matrix. We know that when the data matrix is orthonormal, there is indeed a closed-form solution for the LASSO estimator

$$\hat{\beta}_{L,\lambda} = \begin{bmatrix} \text{sgn}(\hat{\beta}_1)(|\hat{\beta}_1| - \lambda)_+ & \cdots & \text{sgn}(\hat{\beta}_p)(|\hat{\beta}_p| - \lambda)_+ \end{bmatrix}^T,$$

where $\hat{\beta}$ denotes the solution of the ordinary least square problem (Gauraha, 2018). Note that $\hat{\beta}_{L,\lambda}$ is implicitly a function of the response y. Furthermore, Tibshirani and Wasserman (2015) provides a derivation that shows the divergence term in the SURE expression equals the number of predictors whose corresponding regression coefficient is nonzero (i.e. $\|\hat{\beta}_{L,\lambda}\|_0$). As a result, we can write the SURE of the LASSO estimator on a set of PCs as

$$\hat{R}(\lambda) = -n\sigma^2 + \|y - \hat{\beta}_{L,\lambda}(y)\|^2 + 2\sigma^2\|\hat{\beta}_{L,\lambda}\|_0.$$

It now remains to find the optimal λ and subsequently the set of PCs with nonzero regression coefficients. Since each dimension of the LASSO estimator, denoted $\hat{\beta}_{L,\lambda,i}$ is non-smooth at $\hat{\beta}_{L,\lambda,i} = \lambda$, we can use subgradient methods in place of regular gradient descent to obtain the optimal λ (Shor, 2012). Instead of selecting the top k PCs to perform PCR, where k is chosen somewhat arbitrarily, an SURE-inspired variable selection procedure proposed here may help achieve a better quality PCR fit with a similar level of reduction in computational cost.

2.2 Proposal 2: Efficient cross-validation for ridge regression via data subsampling

2.2.1 Motivation and problem formulation

Following the notation from the previous sections, given a data matrix $X \in \mathbb{R}^{n \times d}$, a response vector $y \in \mathbb{R}^n$, and a regularization parameter $\lambda \geq 0$, the ridge estimator of the regression coefficients takes the form

$$\hat{\beta}_{\text{ridge},\lambda}(y) = \left(X^T X + \lambda I\right)^{-1} X^T y,$$

which has a computational complexity of $O(np^2)$. In the context of model selection by minimizng the SURE, one way to obtain the optimal regularization parameter is to run the gradient descent algorithm with respect to λ with the SURE being the objective function. This requires, at each iteration of the optimization, evaluating the derivative of SURE with respect to λ, which involves the term $\hat{\beta}_{\text{ridge},\lambda}(y)$. Therefore, for K steps of gradient descent, the computational complexity is at least in the order of $O(Knp^2)$. In the large-data regime where there are many observations, the cost of this procedure grows linearly with the number of optimization iterations, which can potentially be extremely expensive. Similarly, in the perhaps more commonly used K-fold cross-validation procedure, we need to compute $\hat{\beta}_{\text{ridge},\lambda}$ for each of the K folds of the data. Then if K is also large (e.g. leave-one-out cross-validation), the computational cost could also be expensive.

To reduce the computational complexity of such procedures, one approach is to use a representative subsample of size m ($m \ll n$) in place of the full datase to perform model selection. If the subsample is indeed representative of the full dataset, we can greatly reduce the computational cost without greatly hindering the quality of the selected model. Existing approaches for subsample selection include leverage score based and volume sampling based methods. More specifically, one can either use the leverage scores to form an importance distribution from which we sample of the observations, or use the idea of volume sampling to select a subsample that minimizes the Frobenius norm between the original data matrix and the projection onto the space spanned by the selected subset (Avron et al., 2010; Ma et al., 2014).

It is worth noting that these methods mentioned above do not take into consideration the subsequet model selection step while constructing the subset. While there has been some work that concerns the generalization error of the ridge estimator based on a subset of the data, their result only holds for λ values that are bounded by some function of the true regression coefficients, which are not known a priori (Dereziński and Warmuth, 2017).

2.2.2 Proposed approach

In the context of model selection for ridge regression, ideally we would like a subsample of the data that well approximates the original dataset for a wide range of λ values, or at least the likely λ values given the full dataset. From a probabilistic point of view, we know that the ridge estimator is the MAP estimator with Gaussian likelihood for the data and Gaussian

prior on the regression parameters (with the variance of each dimension being λ^{-1}):

$$\hat{\beta}_{\text{ridge},\lambda} = \arg\max_{\beta} -\frac{1}{2}\sum_{i=1}^{n}(x_i^T\beta - y_i)^2 - \frac{\lambda}{2}\|\beta\|_2^2.$$

From a Bayesian perspective, the Gaussian likelihood on the data and Gaussian prior on the regression parameters together give us a posterior distribution of the regression parameters upon observing the data. Note that the above formulation is still specific to a particular λ value. Upon imposing another prior distribution over λ, we can obtain a posterior distribution π over λ and β. If we can build a sparse and potentially weighted subset (of size m) of the full dataset that has low error across the high density regions of π, it is possible that we can obtain good generalization properties of the ridge estimator resulted from either the SURE or cross-validation model selection procedure.

One way to formulate this idea is as follows. Given an approximation of the posterior distribution over λ and β, which we denote $\hat{\pi}$, we would like to find a set of weights $w \in \mathbb{R}_+^n$, $\|w\|_0 \leq m$, that minimizes

$$\left\| \sum_{i=1}^{n} \mathcal{L}_i - \sum_{i=1}^{n} w_i \mathcal{L}_i \right\|_{\hat{\pi},2},$$

where $\mathcal{L}_i$ is the Gaussian log likelihood for the i^{th} observation, and $\|\cdot\|_{\hat{\pi},2}$ denotes a weighted L^2 norm on the log likelihoods. It turns out that this is precisely the objective for one of the Bayesian coreset construction algorithms, where we can construct a sparse, weighted subsample of the original dataset (Campbell and Broderick, 2019). Since we know that there exists closed-form solutions for weighted least squares, we can use this Bayesian coreset to construct our ridge regression models and to subsequently perform model selection either through SURE or cross-validation (Strutz, 2011).

In Campbell and Broderick (2019), a bound on the above objective using the set of weights obtained through random projection (Rahimi and Recht, 2007) along with samples from $\hat{\pi}$ is provided. In particular, this bound is funciton of m that decreases as m increases. Given the connection between the ridge estimator and its underlying posterior distribution, we should expect to see good generalization properties of the selected model constructed using Bayesian coresets.

In order to construct Bayesian coresets through the above formulation, we need to be able to take samples from $\hat{\pi}$. We note that ridge regression corresponds to a posterior distribution with a Gaussian likelihood on the data and a Gaussian prior on the regression coefficients. Therefore, with an appropriate choice of the prior on λ (e.g. Gaussian), it should not be difficult to obtain a $\hat{\pi}$ that well approximates the true posterior. One option to construct $\hat{\pi}$ is to use the Laplace approximaiton (Bishop and Nasrabadi, 2006). It remains now to explore the effect that different choices of the prior on λ has on the quality of the selected ridge regression model, either through SURE or cross-validation.

3 Project report

In this section, we compare the model selection procedure for ridge regression (i.e. selecting the regularization parameter, discussed in Section 1.1) using SURE as the objective against those based on k-fold cross-validation. We also include ordinary least squares (OLS) in our comparison as a baseline, representing the case of no regularization ($\lambda = 0$). The Python code used to run the experiments and generate the figures can be found at https://github.com/NaitongChen/QP-1.

3.1 Problem setup

We begin by formulating both the SURE and k-fold cross-validation model selection procedures. Recall in our setting of a linear regression problem, we have $y \sim \mathcal{N}(X\beta, \sigma^2 I)$, where $X \in \mathbb{R}^{n\times p}$, $y \in \mathbb{R}^n$, $\sigma > 0$. Also recall that we assume the data are centred and so an intercept term is not needed. We know that for $\lambda > 0$, the ridge estimate of the regression coefficients take on the form

$$\hat{\beta}_{\text{ridge}} = \left(X^T X + \lambda I\right)^{-1} X^T y,$$

then we can write our fitted values as

$$\hat{y}_\lambda(y) = X\hat{\beta}_{\text{ridge}} = X\left(X^T X + \lambda I\right)^{-1} X^T y.$$

Under this setup, the divergence term can be written as

$$\sum_{i=1}^n \frac{\hat{y}_{\lambda,i}(y)}{\partial y_i} = \sum_{i=1}^n \frac{\partial}{\partial y_i}\left(X_i^T\left(X^T X + \lambda I\right)^{-1} X^T y\right) = \text{tr}\left(X\left(X^T X + \lambda I\right)^{-1} X^T\right) = \sum_{j=1}^p \frac{d_j^2}{d_j^2 + \lambda},$$

where the last term is obtained through the singular value decomposition $X = UDV^T$, where D is a diagonal matrix with the singular values $\begin{bmatrix} d_1 & \cdots & d_p \end{bmatrix}$ on the diagonal. We can now write our SURE as

$$\begin{aligned}
\hat{R}(\lambda) &= -n\sigma^2 + \|y - \hat{y}_\lambda(y)\|_2^2 + 2\sigma^2 \sum_{i=1}^n \frac{\hat{y}_{\lambda,i}(y)}{\partial y_i} \\
&= -n\sigma^2 + \|y - \hat{y}_\lambda(y)\|_2^2 + 2\sigma^2 \sum_{j=1}^p \frac{d_j^2}{d_j^2 + \lambda} \\
&= -n\sigma^2 + \|y - \hat{y}_\lambda(y)\|_2^2 + 2\sigma^2 \text{edf}(\lambda).
\end{aligned}$$

Note that $\text{edf}(\lambda)$, the effective degrees of freedom, characterizes the complexity of the model. As λ increases, while we reduce the complexity of the model, the sum of squares residual error reflected through $\|y - \hat{y}_\lambda(y)\|_2^2$ will increase. The SURE then reflects this balance of the bias-variance trade-off. To select λ, we can minimize $\hat{R}$ over λ using gradient descent and automatic differentiation. Note that since we require $\lambda > 0$, we work in the unconstrained parameter space by applying a log transformation to λ.

For k-fold cross-validation, we begin by dividing the dataset into k (almost) equal parts

of size $N_1, \ldots, N_k$ s.t. $\sum_{i=1}^{k} N_i = N$. We denote the index set of each fold as $\mathcal{I}_{N_1}, \ldots, \mathcal{I}_{N_k}$. Given each fold, we compute the mean square prediction error (MSPE) using the regression coefficients estimated with data from all other folds. We pick the regularization parameter λ that minimizes the sum of MSPEs across all folds. We can write the k-fold cross-validation procedure as

$$L(\lambda) = \sum_{i=1}^{k} \frac{1}{N_i} \sum_{j \in \mathcal{I}_{N_i}} \left(y_j - x_j^T \hat{\beta}_{\text{ridge},j}(\lambda) \right)^2,$$

where

$$\hat{\beta}_{\text{ridge},j}(\lambda) = \arg\min_{\beta \in \mathbb{R}^p} \frac{1}{N - N_j} \sum_{l \notin \mathcal{I}_{N_j}} (x_l^T \beta - y_l)^2 + \lambda \|\beta\|_2^2.$$

Note that for each λ, evaluating the loss requires fitting k ridge regression models. In the special case where $k = N$, namely leave-one-out cross-validation (LOOCV), the above loss simplifies to

$$L_{\text{LOOCV}}(\lambda) = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{y_i - \hat{\beta}_{\text{ridge}}(\lambda)}{1 - H_{\lambda,i}} \right)^2,$$

where $\hat{\beta}_{\text{ridge}}$ is the ridge regression estimate using the entire dataset, and $H_{\lambda,i}$ is the i^{th} diagonal entry of $H_\lambda = X(X^T X + \lambda I)^{-1} X^T$.

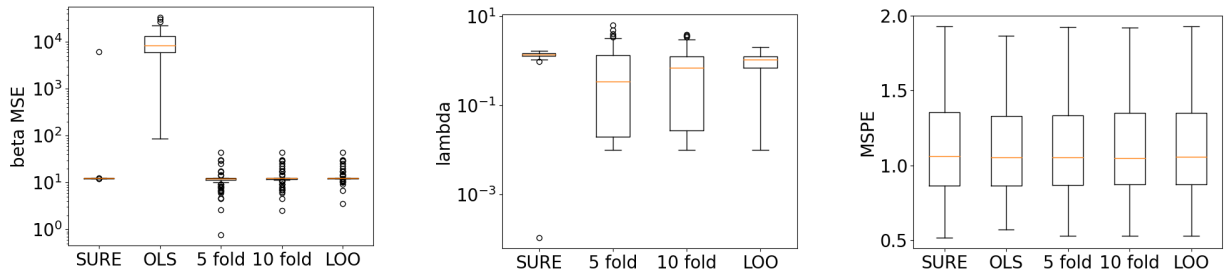
While minimizing $L(\lambda)$ using automatic differentiation and gradient descent is feasible, except for LOOCV, for each optimization iteration, we are required to fit k ridge regression models, making this procedure computationally expensive. As a result, we follow the standard approach of selecting λ over a grid of values. It is also important to note that for $k < N$, the k-fold cross-validation procedure is random over the fold assignment.

3.2 Experiments

Recall that ridge regression is developed as a method to estimate the coefficients in a linear regression problem where the predictor variables are highly correlated. More specifically, by introducing the regularization parameter λ, we sacrifice the unbiasedness of the least squares solution in order to reduce the variance of the estimator of the regression coefficients. As a result, we use a synthetic regression problem with highly correlated predictor variables to evaluate the performance of the model selected using SURE compared to k-fold CV and OLS.

We generate the data for our synthetic regression problem ($p = 5$) consisting of $n = 100$ observations as described below. For each row of the data matrix $\tilde{X}$, we generate

$$\begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \\ \tilde{X}_3 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} -5 \\ 0 \\ 3 \end{bmatrix}, I \right), \quad \tilde{X}_4 = -5\tilde{X}_2 + \mathcal{N}(0, 0.1), \quad \tilde{X}_5 = \tilde{X}_3 + \mathcal{N}(0, 0.1)$$



(a) $\|\hat{\beta} - \beta\|_2^2$ (b) λ (c) estimated MSPE

Figure 1: Comparison of the β MSEs, selected λs, and estimated MSPEs using SURE, OLS, 5 fold cross-validation, 10 fold cross-validation, and LOOCV.

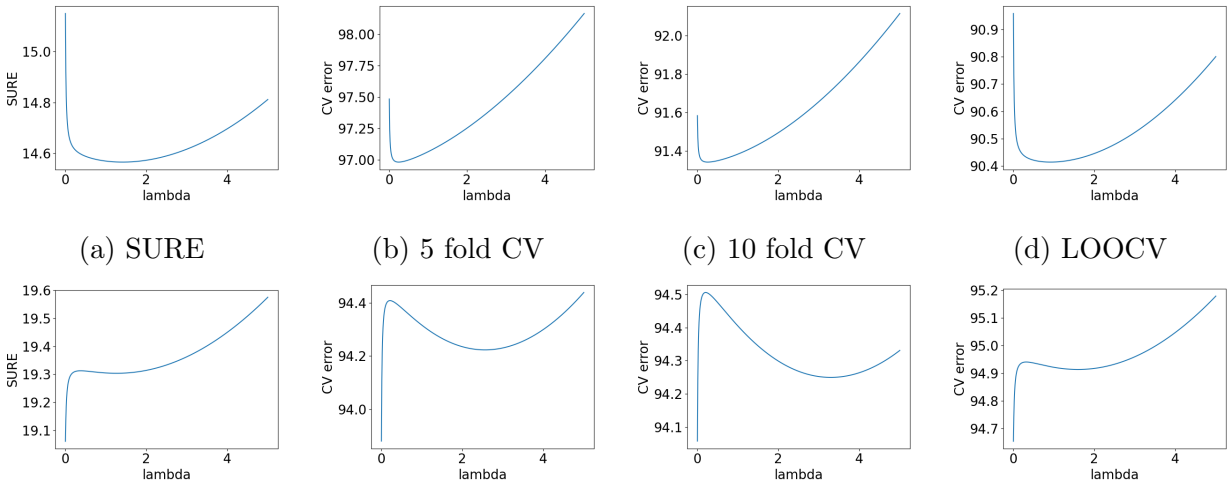
We standardize the features so that each column has mean 0 and variance 1. Denote this transformed data matrix X. The response vector y are then generated by

$$y = X\beta + \epsilon, \quad \text{where} \quad \beta = \begin{bmatrix} 0 & 3 & -1 & 1 & 2 \end{bmatrix}^T, \quad \text{and} \quad \epsilon \sim \mathcal{N}(0, I).$$

Since we know the treu regression coefficients, we can use the mean-squared error between the estimated and true regression coefficients as a metric, in addition to the MSPE, to check how well each method performs. To better assess the performance on average, we randomly split the dataset with 75 observations in the training set and 25 in the test set and compare the models across all training and test set splits.

Fig. 1a shows the side-by-side box plot of the MSE betweeen the true regression coefficients β and the estimated regression coefficients $\hat{\beta}$ corresponding to each of the five procedures (SURE, OLS, 5 fold cross-validation, 10 fold cross-validation, and LOOCV) across 100 trials. We see that OLS has, on average, orders of magnitudes higher errors than the selected model from all over procedures. This is to be expected due to the strong correlations present between pairs of predictor variables in the data. On the other hand, we see that the average β MSE between SURE and all other cross-validation methods are similar, with the variance between trials for SURE being much smaller than others. This variance can be explained by the different λs selected across different trials, as shown in Fig. 1b, where the value of the regularization parameter changes more drastically for cross-validation methods.

Fig. 1c shows the MSPEs for all methods across different trials. Given the similar β MSEs, it makes sense that the MSPEs are similar between SURE and all cross-validation methods. It is worth, however, discussing why the MSPE for OLS is also similar to the models selected using other methods. This can be explained using the relationship between the third (X_3) and fifth (X_5) predictor variables from our synthetic regression problem, where $X_5 \approx X_3$. Since we know the true regression coefficients for these two predictor variables are $\beta_3 = -1$ and $\beta_5 = 2$, respectively, as long as the estimated coefficients $\hat{\beta}_3 + \hat{\beta}_5 \approx -1 + 2 = 1$, we should not expect a much different MSPE. In this particular problem, while OLS does a worse job of recovering the true regression coefficients, it maintains $\hat{\beta}_3 + \hat{\beta}_5 \approx 1$, thus producing a similar MSPE compared to other methods.



(a) SURE (b) 5 fold CV (c) 10 fold CV (d) LOOCV

(a) SURE (b) 5 fold CV (c) 10 fold CV (d) LOOCV

Figure 2: Comparison of the model selection objectives as a function of λs over two random seeds (one seed per row).

3.3 Discussion

To summarize, model selection for ridge regression using SURE in conjunction with automatic differentiation and gradient descent produces comparable results compared to cross-validation based methods. More specifically, it provides a set of estimated regression coefficients closer to the true regression coefficients than OLS in the presence of strong correlation between predictor variables. As a result, in such settings, SURE may be a preferred method for selecting a regularization parameter for ridge regression, as there is no need to fit multiple ridge regression models for each value of the regularization parameter, and that there is no need to specify a grid of regularization parameter values to search over.

Before concluding the report, we discuss a few observations from running the synthetic regression problem. First, for some of the 100 trials, the SURE objective, which estimates a squared loss term, is negative. Although the selected regularization parameter in such cases are still similar to those of the cross-validation based methods, it is worth investigating further to identify the cause of this negative objective function. A possible contributor to this problem is the numerical instability caused by the matrix inversion involved in each step where we solve for the ridge regression coefficients.

Another observation concerns the convexity of the objective functions for all model selection procedures discussed. As noted in Stephenson et al. (2021), it is possible that the objective function for selecting the ridge regularization parameter through cross-validaiton may not be convex. Fig. 2 shows, for two randomly selected training sets, the objectives for both the SURE and cross-validation as a function of the regularization parameter λ. While the objective functions in most of the 100 trials are convex in λ, we see that for certain training sets, they could be nonconvex. It is important to note that the observations from these

training sets are generated from the same underlying distributions. This makes selecting λ using automatic differentiation and gradient descent more challenging. However, it is worth noting that the shape of the objectives between SURE and LOOCV are very similar. We can then possibly leverage the insights from Stephenson et al. (2021) on LOOCV to gauge whether the objection is convex in λ, and thus decide whether to introduce more complicated optimization algorithms other than gradient descent to deal with the nonconvexity.

References

Haim Avron, Petar Maymounkov, and Sivan Toledo. Blendenpik: Supercharging lapack's least-squares solver. *SIAM Journal on Scientific Computing*, 32(3):1217–1236, 2010.

James Berger. Minimax estimation of location vectors for a wide class of densities. *The Annals of Statistics*, pages 1318–1328, 1975.

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Trevor Campbell and Tamara Broderick. Automated scalable bayesian inference via hilbert coresets. *The Journal of Machine Learning Research*, 20(1):551–588, 2019.

Isabella Cera, Laura Whitton, Gary Donohoe, Derek W Morris, Georg Dechant, and Galina Apostolova. Genes encoding satb2-interacting proteins in adult cerebral cortex contribute to human cognitive ability. *PLoS Genetics*, 15(2):e1007890, 2019.

Morris H DeGroot. *Optimal statistical decisions*. John Wiley & Sons, 2005.

Michał Dereziński and Manfred K Warmuth. Subsampling for ridge regression via regularized volume sampling. *arXiv preprint arXiv:1710.05110*, 2017.

Lei Ding, Gabriel E Zentner, and Daniel J McDonald. Sufficient principal component regression for pattern discovery in transcriptomic data. *Bioinformatics advances*, 2(1): vbac033, 2022.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014.

David L Donoho and Iain M Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the american statistical association*, 90(432):1200–1224, 1995.

Niharika Gauraha. Introduction to the lasso. *Resonance*, 23(4):439–464, 2018.

Tom Harel, Naama Peshes-Yaloz, Eran Bacharach, and Irit Gat-Viks. Predicting phenotypic diversity from molecular and genetic data. *Genetics*, 213(1):297–311, 2019.

H Malcolm Hudson. A natural identity for exponential families with applications in multiparameter estimation. *The Annals of Statistics*, 6(3):473–484, 1978.

William James and Charles Stein. Estimation with quadratic loss. In *Breakthroughs in statistics*, pages 443–460. Springer, 1992.

Ian T Jolliffe. A note on the use of principal components in regression. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 31(3):300–303, 1982.

Ker-Chau Li. Asymptotic optimality of cl and generalized cross-validation in ridge regression with application to spline smoothing. *The Annals of Statistics*, pages 1101–1112, 1986.

Florian Luisier, Thierry Blu, and Michael Unser. Image denoising in mixed poisson–gaussian noise. *IEEE Transactions on image processing*, 20(3):696–708, 2010.

Ping Ma, Michael Mahoney, and Bin Yu. A statistical perspective on algorithmic leveraging. In *International Conference on Machine Learning*, pages 91–99. PMLR, 2014.

C. L. Mallows. Some comments on cp. *Technometrics*, 15(4):661–675, 1973. ISSN 00401706. URL http://www.jstor.org/stable/1267380.

Christopher A Metzler, Ali Mousavi, Reinhard Heckel, and Richard G Baraniuk. Unsupervised learning with stein's unbiased risk estimator. *arXiv preprint arXiv:1805.10531*, 2018.

Bala Kishore Panisetti, Thierry Blu, and Chandra Sekhar Seelamantula. An unbiased risk estimator for multiplicative noise—application to 1-d signal denoising. In *2014 19th International Conference on Digital Signal Processing*, pages 497–502. IEEE, 2014.

James Chao-Ming Peng. *Simultaneous estimation of the parameters of independent Poisson distributions*. Stanford University, 1975.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.

Sathish Ramani, Thierry Blu, and Michael Unser. Monte-carlo sure: A black-box optimization of regularization parameters for general denoising algorithms. *IEEE Transactions on image processing*, 17(9):1540–1554, 2008.

Naum Zuselevich Shor. *Minimization methods for non-differentiable functions*, volume 3. Springer Science & Business Media, 2012.

Charles Stein. Inadmissibility of the usual estimator of the mean of a multivariate normal distribution. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability: Contributions to the Theory of Statistics*, volume 1, page 197. University of California Press, 1956.

Will Stephenson, Zachary Frangella, Madeleine Udell, and Tamara Broderick. Can we globally optimize cross-validation loss? quasiconvexity in ridge regression. *Advances in Neural Information Processing Systems*, 34:24352–24364, 2021.

Tilo Strutz. *Data fitting and uncertainty: A practical introduction to weighted least squares and beyond*. Springer, 2011.

Ryan Tibshirani and L Wasserman. Stein's unbiased risk estimate. *Course notes from "Statistical Machine Learning*, pages 1–12, 2015.

Kam-Wah Tsui. *Simultaneous estimation of the parameters of the distributions of independent Poisson random variables*. PhD thesis, University of British Columbia, 1978.

Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6721–6729, 2017.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017.