JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2022, VOL. 117, NO. 539, 1516–1529: Theory and Methods
https://doi.org/10.1080/01621459.2020.1864380

Covariate Information Number for Feature Screening in Ultrahigh-Dimensional Supervised Problems

Debmalya Nandy[a], Francesca Chiaromonte[b,c], and Runze Li[b]

[a]Department of Biostatistics & Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, CO; [b]Department of Statistics, Penn State University, University Park, PA; [c]Institute of Economics and EMbeDS, Sant'Anna School of Advanced Studies, Pisa, Italy

ABSTRACT

Contemporary high-throughput experimental and surveying techniques give rise to ultrahigh-dimensional supervised problems with sparse signals; that is, a limited number of observations (n), each with a very large number of covariates ($p \gg n$), only a small share of which is truly associated with the response. In these settings, major concerns on computational burden, algorithmic stability, and statistical accuracy call for substantially reducing the feature space by eliminating redundant covariates before the use of any sophisticated statistical analysis. Along the lines of *Pearson's correlation coefficient-based sure independence screening* and other model- and correlation-based feature screening methods, we propose a model-free procedure called *covariate information number-sure independence screening* (CIS). CIS uses a marginal utility connected to the notion of the traditional Fisher information, possesses the sure screening property, and is applicable to any type of response (features) with continuous features (response). Simulations and an application to transcriptomic data on rats reveal the comparative strengths of CIS over some popular feature screening methods. Supplementary materials for this article are available online.

ARTICLE HISTORY
Received November 2018
Accepted December 2020

KEYWORDS
Affymetrix GeneChip Rat Genome 230 2.0 Array; Fisher information; Model-free; Supervised problems; Sure independence screening; Ultrahigh dimension

1. Introduction

Contemporary high-throughput experimental and surveying techniques employed in many scientific fields often generate data on an enormous number of variables. This is the case, for example, in the "Omics" sciences (Genomics, Transcriptomics, Proteomics, Metabolomics, etc.) and in biomedical applications involving imaging and/or the analysis of extensive electronic medical records. Extracting meaningful and interpretable information from these data often requires studying the association of a response with thousands to millions of predictors (p)—here and in the following, we use "feature," "predictor," and "covariate" interchangeably, all indicating a potential explanatory variable for the response. Even when these supervised problems have what appears to be a large sample size (n), this can in fact be orders of magnitude smaller than p. For instance, in transcriptomic studies n may be in the hundreds, but p may be in the thousands or tens of thousands. This is commonly referred to as an *ultrahigh-dimensional* setting, and often linked to the fact that p increases as a function of n (collecting more observations produces more features). Roughly speaking, one can have $p \approx \exp\{\mathcal{O}(n^{\xi})\}, \xi > 0$ (Fan and Lv 2008). Importantly, in ultrahigh-dimensional settings, association signals are often sparse, that is, only a handful of predictors contribute to explaining variation in the response.

When p exceeds n, computing or inverting sample covariance matrices ($\widehat{\Sigma}$) to estimate dependencies among predictors becomes very inaccurate, numerically unstable, or downright unfeasible (Ledoit and Wolf 2004; Schäfer and Strimmer 2005; Bickel and Levina 2008; Chen, Chi, and Goldsmith 2015). This, in turn, negatively affects regression model fitting, classification methods, and also techniques for supervised dimension reduction—in their standard versions, most of these tools employ some versions of $\widehat{\Sigma}$ or $\widehat{\Sigma}^{-1}$.

Popular methods such as LASSO (Tibshirani 1996), SCAD (Fan and Li 2001), Elastic Net (Zou and Hastie 2005), and MCP (Zhang 2010) use penalties to regularize supervised problems, performing feature selection and model fitting simultaneously (see Fan and Lv 2010 for a comprehensive overview). In practice, many of these methods may successfully handle $p > n$ scenarios, but they also deteriorate and might not scale up in realistic time when $p \gg n$ (see, e.g., Fan and Lv 2008, Table 1, p. 862). Fan, Fan, and Wu (2011) further discussed the challenges of high and ultrahigh dimension in classification. "Curse of dimensionality" issues including computational burden, statistical inaccuracy, and algorithmic instability call for alternative approaches in tackling ultrahigh-dimensional supervised problems (Fan, Samworth, and Wu 2009). One such approach is *feature screening*, pioneered by the development of *sure independence screening* (SIS; Fan and Lv 2008) and extensively studied ever since, it led to *model-based*, *model-free*, *correlation-based*, and *distance-based* procedures for a variety of supervised problems including regression, classification, discriminant analysis, survival analysis, etc. Liu, Zhong, and Li (2015) provided a comprehensive overview of work prior to 2015 (we also provide a

CONTACT Debmalya Nandy debmalya.nandy@cuanschutz.edu Department of Biostatistics & Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, CO 80045.

Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JASA.

selected list of references in Section S5 of the supplementary materials).

In this article, we propose *covariate information number-sure independence screening* (CIS), a model-free feature screening procedure based on a novel marginal utility called *covariate information number* (CIN). For each covariate X_j, the CIN captures marginal association with the response Y without assuming any specific underlying model, and can be interpreted in terms of the traditional Fisher information in Statistics. CIN is computed from the joint density of (Y, X_j) employing kernels to estimate marginal and *inverse* conditional densities within response sub-populations that are naturally defined when Y is categorical or discrete, and artificially generated by slicing the range if Y is continuous. Thus, our approach can be employed irrespective of the nature of the response and, in fact for continuous Y, it is robust to outliers (it uses only the ranks of the Y values for slicing). Moreover, switching the roles of Y and X_j in computing the CIN, our approach can be utilized also to screen discrete or categorical covariates—as long as the response is continuous. Because of the way information on marginal and conditional densities is used in the CIN calculation, in addition to being model-free, our approach does not require strong assumptions on the predictors (the rationale is similar to that discussed in Yao et al. (2019)). Under mild regularity conditions, we show that the CIS procedure built upon the CIN marginal utility possesses the *sure screening property* (Fan and Lv 2008). Overall, we find that CIS is competitive with, and sometimes better than, other popular feature screening procedures.

We compare CIS to five other procedures: (a) SIS (Fan and Lv 2008), (b) *high-dimensional ordinary least squares projection* (HOLP; Wang and Leng 2016), (c) *sure independent ranking and screening* (SIRS; Zhu et al. 2011), (d) *distance correlation-sure independence screening* (DC-SIS; Li, Zhong, and Zhu 2012), and (e) *martingale difference correlation-sure independence screening* (MDC-SIS; Shao and Zhang 2014). SIS postulates a naive linear regression model, and screens the predictors based on the magnitudes of their *Pearson correlations* with the response. It is intuitive, computationally straightforward, and possesses the sure screening property. SIRS, DC-SIS, and MDC-SIS are among the most popular model-free feature screening procedures in the literature. Specifically, SIRS does not postulate a specific underlying model but relies instead on a general framework which includes many common parametric and semi-parametric models. It possesses *rank consistency*, a stronger property than sure screening, and allows both univariate and multivariate responses. Moreover, similar to our CIS, it only considers response ranks and is therefore robust to outliers. DC-SIS screens the predictors based on their *distance correlations* (Székely, Rizzo, and Bakirov 2007) with the response—a measure of departure from independence for two random vectors, built through characteristic functions. It possesses the sure screening property, allows both univariate and multivariate responses, and can also handle grouped predictors. MDC-SIS uses *martingale difference correlations*, which are a natural extension of distance correlations. It screens predictors that contribute to the *conditional mean* of $Y|\mathbf{X}$, and has the sure screening property. Notably, to tackle regressions with heteroscedastic errors, the authors of MDC-SIS proposed an extension that screens based on contributions to *conditional quantiles*. While the feature screening procedures discussed above involve computing the marginal utilities independently, HOLP involves a joint estimation of these measures. HOLP is motivated by the *ordinary least squares* estimator and *ridge regression*, straightforward and efficient to compute, and relaxes the often violated assumption of strong marginal correlations between each truly associated covariate and the response. However, similar to SIS, HOLP also postulates a naive linear regression model. All procedures in (a)–(e), as well as CIS, allow p to grow exponentially with n.

The rest of the article is organized as follows. Section 2 contains the details of our proposal—formulation, properties, and implementation of the CIN; the CIS algorithm; and the theoretical properties of CIS under appropriate assumptions. Section 3 presents an extensive simulation study to compare the performance of CIS to those of the five popular screening procedures mentioned above. Section 4 presents an application to transcriptomic data on Norway rats (GeneChip Rat Genome 230 2.0 Array Data; Scheetz et al. 2006). Concluding remarks are provided in Section 5, whereas proofs of theoretical results, full simulation results, details on the transcriptomic data application, and some relevant additional information are provided in an online supplementary materials (an "S" in the numbered references below indicates sections, tables, and figures in the supplementary materials).

2. CIN—Sure Independence Screening (CIS)

In this section, we describe our proposal. We introduce the general setup, provide details on the formulation, properties, and implementation of our CIN marginal utility, describe our CIN-based screening algorithm (CIS) and discuss its theoretical properties under appropriate assumptions. Notation is similar to that in Zhu et al. (2011).

Consider a univariate response Y with support Θ_Y and a p-dimensional covariate vector $\mathbf{X} = (X_1, \ldots, X_p)^T$ with $p \gg n$. Let $F(y|\mathbf{x}) = P(Y \leq y|\mathbf{x})$ denote the conditional distribution of Y given $\mathbf{X} = \mathbf{x}$ in the following definitions of the two index sets:

$$\begin{aligned}\mathcal{A} &= \{j : F(y|\mathbf{x}) \text{ is functionally dependent on } X_j \text{ for some } y \in \Theta_Y\}\\ \mathcal{I} &= \{j : F(y|\mathbf{x}) \text{ is not functionally dependent on } X_j \text{ for any } y \in \Theta_Y\}\\ &= \{1, 2, \ldots, p\} \setminus \mathcal{A}.\end{aligned}$$

$\mathcal{A}$ indexes predictors that are truly associated with the response; it is called the *active* set. $\mathcal{I}$ indexes the remaining, *inactive* predictors. Note that, in this definition, $F(y|\mathbf{x})$ is completely generic—no model form is specified. Let $s = |\mathcal{A}|$ denote the cardinality of $\mathcal{A}$; s out of p covariates are active, and therefore, measures the sparsity level of the association between Y and $\mathbf{X}$. In any feature screening procedure, including our CIS, the objective is to estimate $\mathcal{A}$ conservatively; that is, with a minimal prevalence of *false negatives*.

2.1. CIN

Next, we expand upon the CIN, the marginal utility for our proposed CIS. Some of the developments follow directly as

special cases ($p = 1$) of those for the *covariate information matrix* (Yao et al. 2019).

Let $f(y|x_j)$ and $f_j(x_j)$, respectively, denote the conditional density of Y given $X_j = x_j$ and the marginal density of X_j, and assume that the standard regularity conditions for likelihood analysis hold (see Section 2.6.1). Treating the observed x_j as a "parameter," we can use the traditional *Fisher information* formulation to create the quantity

$$\mathbb{F}_{x_j} = \int \left[\frac{\partial}{\partial x_j} \log(f(y|x_j)\right]^2 f(y|x_j)dy.$$

Capturing the information that $f(y|x_j)$ would carry about x_j, if it were in fact unknown, this provides a local measure of association. Based on the bivariate joint distribution of (Y, X_j), the CIN for covariate X_j is then defined as the expected value of this quantity; that is

$$\omega_j = \int \mathbb{F}_{x_j} f_j(x_j)dx_j. \tag{1}$$

Estimates of the scalars $\omega_j, j = 1, 2 \ldots, p$, which are theoretically nonnegative by definition, are key components of the final form of the marginal utilities (see below) we use for ranking the covariates in our CIS screening procedure.

We next introduce two other quantities which are relevant to our proposal. Here, we need to also consider $f_j(x_j|y)$ and $f(y)$, the *inverse* conditional density of X_j given $Y = y$ and the marginal density of Y, again with standard regularity conditions (see Section 2.6.1). The *density information* (Hui and Lindsay 2010; Lindsay and Yao 2012; Yao et al. 2019) in the marginal density of X_j is defined as

$$\mathbb{J}_{X_j} = \int \left[\frac{\partial}{\partial x_j} \log(f_j(x_j))\right]^2 f_j(x_j)dx_j. \tag{2}$$

Similarly, the density information in the conditional density of $X_j|Y = y$ is defined as

$$\mathbb{J}_{X_j|Y=y} = \int \left[\frac{\partial}{\partial x_j} \log(f_j(x_j|y))\right]^2 f_j(x_j|y)dx_j,$$

and can be averaged to produce

$$\mathbb{J}_{X|Y} = \int \mathbb{J}_{X|Y=y} f(y)dy. \tag{3}$$

The marginal utility which CIS uses for each covariate X_j, $j = 1, 2, \ldots, p$, is the CIN (ω_j) normalized by its density information, that is, $\omega_j^* = \omega_j/\mathbb{J}_{X_j}$. The next theorem summarizes some properties of ω_j^* (proofs are in Section S1).

Theorem 2.1 (Properties of the normalized CIN).

(i) The normalized CIN $\omega_j^* = 0$ if and only if Y and X_j are statistically independent.

(ii) If (Y, X_j) follows a bivariate normal distribution with correlation coefficient ρ_j, then the normalized CIN ω_j^* is a monotonically increasing function of $|\rho_j|$.

(iii) ω_j can be expressed as $\omega_j = \mathbb{J}_{X_j|Y} - \mathbb{J}_{X_j}$. Hence, the normalized CIN ω_j^* is

$$\omega_j^* = \frac{\mathbb{J}_{X_j|Y} - \mathbb{J}_{X_j}}{\mathbb{J}_{X_j}} = \frac{\mathbb{J}_{X_j|Y}}{\mathbb{J}_{X_j}} - 1. \tag{4}$$

(iv) If $a \neq 0 \in \mathbb{R}$ and $b \in \mathbb{R}$ are two constants and $\tilde{X}_j = aX_j + b$, then the normalized CIN of $\tilde{X}_j$ is $\tilde{\omega}_j^* = \omega_j^*$, the normalized CIN for X_j.

Henceforth, we will ignore the subtraction of 1 in (4) (which has no effect in the ranking of the X_j's). We will consider $\frac{\mathbb{J}_{X_j|Y}}{\mathbb{J}_{X_j}}$ and simply refer to it as the CIN. Properties (i) and (ii) motivate its use as a marginal utility in feature screening: positive values of ω_j^* correspond to statistical dependence between Y and X_j and, in a bivariate Gaussian scenario where the association is linear, ω_j^* increases with the absolute value of the correlation coefficient. Notably, this fact implies that ω_j^* is a more general measure of association than the marginal utility employed by SIS (Fan and Lv 2008)—capturing also potential nonlinear dependencies between Y and X_j. (iii) reformulates the CIN as the ratio of the average density information in $f_j(x_j|y)$ (inverse regression) to the density information in $f_j(x_j)$, the marginal density of the predictor X_j. Following the argument in Yao et al. (2019), the ratio in Equation (4) "cleanses" the association signal from potential distributional peculiarities of X_j, and renders ω_j^* effective also for "not well-behaved" covariates. Finally, (iv) describes the effects of affine transformations on ω_j^*.

Note that (3) can be easily adapted for a discrete or categorical response by simply replacing the integral with an appropriate sum. If $Y \in \{y^{(1)}, \ldots, y^{(L)}\}$ with $\Pr(Y = y^{(\ell)}) = \pi_\ell$, $\ell = 1, \ldots, L$, one has

$$\mathbb{J}_{X_j|Y} = \sum_{\ell=1}^{L} \pi_\ell \mathbb{J}_{X_j|Y=y^{(\ell)}}. \tag{5}$$

The ratio of (5) to (2) provides a straightforward definition of the CIN in discrete regressions or classification problems.

2.2. Estimation of the CIN

Up to this point, we have defined and characterized the CIN theoretically, at the population level. Next, we describe its estimation for the practical implementation of our CIS procedure on sample data.

Three facts are key: First, we write the CIN through (4), which comprises two quantities, $\mathbb{J}_{X_j|Y}$ and $\mathbb{J}_{X_j}$. Second, regardless of the nature of the response, we write $\mathbb{J}_{X_j|Y}$ through its formulation in (5); if the response is continuous, we create an approximate version with "sub-populations" defined by *slicing* the range of Y. Notably, this slicing strategy is used by most sufficient dimension reduction methods based on inverse regression, for example, SIR (Li 1991), SAVE (Cook and Weisberg 1991), SR (Wang and Xia 2008), and CIM (Yao et al. 2019). Third, we estimate $\mathbb{J}_{X_j}$ and the components $\mathbb{J}_{X_j|Y=y^{(\ell)}}$, $\ell = 1, 2, \ldots, L$ of $\mathbb{J}_{X_j|Y}$ through kernels.

Let us drop the predictor index j to simplify notation (X now stands for a generic predictor) and start with the estimation of $\mathbb{J}_X$. (2) expresses $\mathbb{J}_X$ as an expectation: $\mathbb{J}_X = E_X\left[\frac{\partial}{\partial X} \log f(X)\right]^2 = E_X[g(X)]$. We therefore need to estimate the density $f(\cdot)$ and the expectation of the function $g(x) =$

$\left[\frac{\partial}{\partial x}\log f(x)\right]^2$. We use a kernel density estimator

$$f_n(x;h) = \frac{1}{n}\sum_{i=1}^{n} k_h(x - x_i) \text{ with } k_h(t) = \frac{1}{h}k\left(\frac{t}{h}\right), \quad (6)$$

and replace the theoretical expectation by the sample average, which gives $\widehat{\mathbb{J}}_X = \widehat{E}_X[\widehat{g}(X)] = \frac{1}{n}\sum_{i=1}^{n}\widehat{g}(x_i)$. Next, using observations within slices (if Y is continuous) or natural sub-populations (if Y is discrete or categorical), we produce each $\widehat{\mathbb{J}}_{\mathbf{X}|Y=y^{(\ell)}}$, $\ell = 1, 2, \ldots, L$ in exactly the same way, and set $\widehat{\pi}_\ell = \frac{n_\ell}{n}$, $\ell = 1, 2, \ldots, L$ (where n_ℓ is the number of observations with $Y = y^{(\ell)}$). Thus, we estimate

$$\widehat{\mathbb{J}}_{X|Y} = \sum_{\ell=1}^{L}\widehat{\pi}_\ell\widehat{\mathbb{J}}_{X|Y=y^{(\ell)}}. \quad (7)$$

Finally, we compute a ratio to produce: $\widehat{\omega}_j^* = \widehat{\mathbb{J}}_{X|Y}/\widehat{\mathbb{J}}_X$.

An important remark is for the case of a continuous response: slices are customarily produced as to contain (approximately) the same number of observations. Thus, the partitioning does not use the observed values y_i, $i = 1, 2, \ldots, n$, but rather their ranks. Consequently, similar to SIRS (Zhu et al. 2011), our CIS screening is robust to outliers in the response (see Model(4) in Section 3.2).

2.3. Tuning Parameters

Following the description regarding (5) and in Section 2.2, when the response is discrete or categorical, the number of "slices" L is given. However, when the response is continuous, the choice of L is critical. This is a well-recognized challenge in inverse regression-based sufficient dimension reduction methods (see Wang and Xia 2008; Yao et al. 2019). There is a tradeoff between using more slices to achieve a more accurate approximation of the overall object of interest (in our case, $\mathbb{J}_{X|Y}$) and using fewer slices to have a sufficiently large number of observations for in-slice calculations (in our case, the estimation of each $\mathbb{J}_{X|Y=y^{(\ell)}}$). For simulations in Section 3, we used total sample sizes n = 200 and 600 and investigated the CIS screening performance for $L = 2, 3, 5, 8, 10$, and 12. CIS performance does vary with L, because we need a large enough sample size within each slice for kernel density estimation to be reliable. We find that moderate values (e.g., $L = 3$–8) work well in most cases, and that the effect of L becomes negligible as the total n becomes larger and/or the signal to noise ratio in the data becomes stronger (see Sections 3 and S3).

Another critical choice in our estimation is that of *kernel* and *bandwidth*. In our implementation, we use a simple Gaussian kernel for $k_h(\cdot)$ ($k_{h^{(\ell)}}(\cdot)$ for sub-population with $Y = y^{(\ell)}$) and *Silverman's rule of thumb* for the bandwidth, which sets $h = 1.06 \times \widehat{\sigma}_x n^{-1/5}$ ($h^{(\ell)} = 1.06 \times \widehat{\sigma}_{x^{(\ell)}} n_l^{-1/5}$; see Silverman 2018), where $\widehat{\sigma}_x$ ($\widehat{\sigma}_{x^{(\ell)}}$) is the sample standard deviation of X ($X^{(\ell)}$: X-observations with $Y = y^{(\ell)}$).

2.4. Computational Burden

Computational burden is an important consideration for any feature screening procedure. In addition to the number of covariates to be screened (p), it depends on the time needed to calculate each marginal utility. Our CIN marginal utility ($\widehat{\omega}_j^*$) has a reasonable computational cost, making CIS viable also in applications with large number of covariates (see Section 4), and comparable to other model-free screens. For instance, we performed a comparative study on the elapsed computation times for CIS and five additional screening procedures (see Sections 1, 3.2, and 4) for a simulation scenario with p = 2000 and n = 200 (this used Model(3)(a) with $\sigma = 0.5$ and $\Sigma_{\mathbf{X}} = \Sigma_{\mathbf{X}}^{(I)}$; see Section 3.2). In this comparison, all methods except HOLP were implemented using MATLAB (MATLAB 2020), version 9.9.0.1467703 (R2020b). HOLP was implemented in R (R Core Team 2020), version 4.0.2, using the GitHub R package *screening* available at *https://github.com/wwrechard/screening* (see Section S6 for more details). All codes were run on a MacBook Pro 2019 laptop with macOS Mojave Version 10.14.6, 2.3 GHz Intel Core i9 processor, and 16 GB 2400 MHz DDR4 RAM.

Taking the medians over 100 simulation runs, CIS with L = 5 slices took ≈3.054 sec to compute all p = 2000 marginal utilities (see also Table 1). This was higher but comparable to the widely used model-free DC-SIS, that took ≈1.222 sec. As to be expected given the much simpler nature of their marginal utilities, SIS and HOLP were much faster—taking, respectively, ≈0.082 and ≈0.074 sec. SIRS and MDC-SIS, which are also model-free, took, respectively, ≈0.414 and ≈0.741 sec (see Table S25 for run times with n = 600). Note that, except for HOLP, since the utility of each covariate is computed marginally, total computation time scales linearly with p. Extrapolating from the calculations above, in an application with n = 200 and as many as p = 2,000,000 covariates, CIS would compute all marginal utilities in ≈50 minutes. Of course, computation time could be vastly reduced implementing screens in more efficient computer programming languages, such as *C* (Ritchie, Kernighan, and Lesk 1988).

2.5. CIS Algorithm

Let $(y_i, \mathbf{x}_i)$, $i = 1, 2, \ldots, n$ be a random sample from the distribution of $(Y, \mathbf{X})$. To implement CIS in practice, we proceed as follows:

Table 1. Total computation time (seconds; median over 100 runs) required for computing $p = 2000$ marginal utilities (Model(3)(a) with $\Sigma_{\mathbf{X}} = \Sigma_{\mathbf{X}}^{(I)}$, $\sigma = 0.5$, and $n = 200$) for different screening procedures. NOTE: Number of slices (L) for CIS vary as indicated in the table. HOLP was run in R (R Core Team 2020; version 4.0.2) and the rest in MATLAB (MATLAB 2020; version 9.9.0.1467703 (R2020b)).

SIS	HOLP	SIRS	DC-SIS	MDC-SIS	CIS[3]	CIS[5]	CIS[8]	CIS[10]	CIS[12]
0.082	0.074	0.414	1.222	0.741	3.338	3.054	2.966	2.980	3.010

1. For $j = 1, 2, \ldots, p$, compute $\widehat{\mathbb{J}}_{X_j|Y}$, $\widehat{\mathbb{J}}_{X_j}$, and $\widehat{\omega}_j^* = \frac{\widehat{\mathbb{J}}_{X_j|Y}}{\widehat{\mathbb{J}}_{X_j}}$ (Section 2.2).
2. Order $\widehat{\omega}_{(1)}^* \geq \widehat{\omega}_{(2)}^* \geq \cdots \geq \widehat{\omega}_{(p)}^*$ and estimate $\widehat{\mathcal{A}}_d$ as the top d ranking covariates.

The normalization in (1) ensures that all marginal utilities are on the same *density information scale*. Moreover, by Theorem 2.1(iv), the ratio expressing $\widehat{\omega}_j^*$ is invariant to affine transformations of X_j, for example, marginal centering and/or scaling to unit variance.

The calculations in (1) involve the tuning parameters L (number of slices, if the response is continuous) and h (bandwidth used in kernel density estimation) discussed in Section 2.3. Notably, (2) involves another crucial quantity that plays a role possibly much more vital than those of tuning parameters in the marginal utility calculation: the number d of covariates retained in the screen. Following the literature (e.g., Fan and Lv 2008; Li, Zhong, and Zhu 2012; Shao and Zhang 2014), in our simulations we employ a *hard threshold* defined as $d = k \times \lfloor \frac{n}{\log(n)} \rfloor$ with constant multiplier $k = 1, 2$, or 3 (see Sections 3 and S3). However, this is an intriguing open question for screening algorithms. For instance, Zhu et al. (2011) showed that one can use a hard threshold, a *soft threshold*, or a combination of both. How to select d in an effective and data-driven fashion is beyond the scope of this article, but we hint at the development of a potential diagnostics in Section 5.

2.6. Sure Screening Property of CIS

In this section, we establish the *sure screening property* (Fan and Lv 2008) for our CIS, built upon the CIN marginal utility. At the outset, we adjust the definition of the estimated active set as follows:

$$\widehat{\mathcal{A}} = \left\{ j : \widehat{\omega}_j^* \geq c_0 n^{-\kappa}, 1 \leq j \leq p \right\}, \tag{8}$$

where $c_0 > 0$ and $0 < \kappa < \gamma < \frac{1}{3}$ are given constants (see below). In proving the sure screening property, we considered a discrete or categorical Y with L distinct values or labels. Recall that if Y is continuous, we operate with its "discretized" version obtained through slicing (see Sections 2.2 and 2.3).

2.6.1. Assumptions and Regularity Conditions

In proving sure screening, we assume that all active covariates X_j, $j \in \mathcal{A}$ satisfy a *minimum signal strength condition*, and specifically that:

$$\min_{j \in \mathcal{A}} \omega_j^* \geq 2c_0 n^{-\kappa}, \tag{9}$$

where $c_0 > 0$ and $0 < \kappa < \gamma < \frac{1}{3}$ are the same constants that appear in (8). Note that, here the minimum value for the signals (the true marginal utilities) is *twice* $c_0 n^{-\kappa}$. As pointed out in Liu, Li, and Wu (2014), this assumption bounds the marginal utilities of active covariates away from 0 for any finite n. However, as n increases, this minimum signal strength can decrease, converging to 0 asymptotically. This fact indicates that, when n is very large, our procedure with the sure screening property can retain covariates whose marginal association with Y is negligible, but are jointly associated with the response. This assumption corresponds to Condition 3 in Fan and Lv (2008) and is commonly used in the screening literature (see, e.g., Li, Zhong, and Zhu 2012; Shao and Zhang 2014). Also note that, while (9) states the assumption on the minimal signal strength of the active covariates, we do not assume any condition on the order of the maximum signal strength to establish the sure screening property of CIS.

We also assume that the number of Y subpopulations (or slices) L is finite, and impose some regularity conditions on (i) kernel densities and associated bandwidths: $k(\cdot)$, h, $k_{h^{(\ell)}}(\cdot)$, $h^{(\ell)}$, $\ell = 1, \ldots, L$, used for estimating each CIN (see (6) and (7)); (ii) marginal and inverse conditional covariate densities: $f_j(\cdot)$ and $f_j(\cdot|Y = y^{(\ell)})$, $\ell = 1, 2, \ldots, L$; and (iii) marginal and inverse density information: $\mathbb{J}_X$ and $\mathbb{J}_{X|Y=y^{(\ell)}}, \ell = 1, 2, \ldots, L$. These regularity conditions are described below, neglecting again the covariate subscript j for notational simplicity.

Kernel densities:

C1. $k_h(\cdot)$ and $k_{h^{(\ell)}}(\cdot)$, $\ell = 1, 2, \ldots, L$, have bounded variance.
C2. $h = \mathcal{O}(n^{-\gamma})$, $0 < \kappa < \gamma < \frac{1}{3}$; and for each $\ell = 1, 2, \ldots, L$, $h^{(\ell)} = \mathcal{O}(n_\ell^{-\gamma})$, where n_ℓ is the number of observations with $Y = y^{(\ell)}$.
C3. $k_h(\cdot)$ and $k_{h^{(\ell)}}(\cdot)$, $\ell = 1, 2, \ldots, L$, are order-1 kernels with nonvanishing first derivatives (see Section S1.5).
C4. $\sup_{x \in \chi} k_h'(x)$ and $\sup_{x \in \chi} k_{h^{(\ell)}}'(x)$, $\ell = 1, 2, \ldots, L$, are bounded above.
C5. the βth moments ($1 \leq \beta < 2$) of absolute values for the kernel densities $k_h(\cdot)$ and $k_{h^{(\ell)}}(\cdot)$, $\ell = 1, 2, \ldots, L$, are finite.

Covariate densities:

C6. $f^2(\cdot)$ and $f^2(\cdot|Y = y^{(\ell)})$, $\ell = 1, 2, \ldots, L$, are uniformly bounded away from zero.
C7. $f'(\cdot)$ and $f'(\cdot|Y = y^{(\ell)})$, $\ell = 1, 2, \ldots, L$, belong to the Hölder class $\Sigma(\beta, \Lambda)$ where $1 \leq \beta < 2$ and $\Lambda > 0$ are constants (see Section S1.5).
C8. $\sup_{x \in \chi} f(x)$ and $\sup_{x \in \chi} f(x|Y = y^{(\ell)})$, $\ell = 1, 2, \ldots, L$, are bounded above.
C9. $\sup_{x \in \chi} |f'(x)|$ and $\sup_{x \in \chi} |f'(x|Y = y^{(\ell)})|$, $\ell = 1, 2, \ldots, L$, are bounded above.
C10. $\sup_{x \in \chi} \frac{(f'(x))^2}{f^2(x)}$ and $\sup_{x \in \chi} \frac{(f'(x|Y=y^{(\ell)}))^2}{f^2(x|Y=y^{(\ell)})}$, $\ell = 1, 2, \ldots, L$, are bounded above.

Density informations:

C11. $\min_{1 \leq \ell \leq L} \mathbb{J}_{X|Y=y^{(\ell)}}$, and hence $\mathbb{J}_{X|Y}$, are uniformly bounded away from zero.
C12. $\mathbb{J}_X$, $\max_{1 \leq \ell \leq L} \mathbb{J}_{X|Y=y^{(\ell)}}$, and hence $\mathbb{J}_{X|Y}$, are uniformly bounded above (from C10).
C13. $\mathbb{J}_X$ and $\widehat{\mathbb{J}}_X$ are bounded away from zero.

In the context defined by the assumptions and conditions above, we have the following theorem (additional details and proofs are provided in Sections S1.6–S1.8).

Theorem 2.2 (Sure screening property for CIS). Let ξ_0^* and c_0 be positive constants, κ and γ be constants such that $0 < \kappa < \gamma < \frac{1}{3}$, and $n_{(1)} = \min_{1\le\ell\le L} n_\ell$ be the size of the least numerous class (or slice). For $j = 1, 2, \ldots, p$, we have

$$P\left(\max_{1\le j\le p} |\widehat{\omega}_j^* - \omega_j^*| > c_0 n^{-\kappa}\right) \le \mathcal{O}\left(np\exp\left(-\frac{n^{-\kappa} n_{(1)}^{\gamma}}{\xi_0^*}\right)\right),$$

where ω_j^* and $\widehat{\omega}_j^*$ are the true and the estimated CIN marginal utilities, respectively. Moreover, using the definition of $\widehat{\mathcal{A}}$ in (8) and assuming the minimum signal strength condition in (9), we have

$$P\left(\mathcal{A} \subset \widehat{\mathcal{A}}\right) \ge 1 - \mathcal{O}\left(ns_n\exp\left(-\frac{n^{-\kappa} n_{(1)}^{\gamma}}{\xi_0^*}\right)\right), \qquad (10)$$

where s_n is the cardinality of the active set $\mathcal{A}$.

Note that the cardinality of $\mathcal{A}$ in (10), s_n, is indexed as to indicate dependence on n: CIS guarantees sure screening when the number of covariates (p) as well as the number of active covariates grow as we gather more observations (see also Li, Zhong, and Zhu 2012; Liu, Li, and Wu 2014; Shao and Zhang 2014). Concerning the way p grows with the sample size, the exponent in (10) shows that CIS guarantees sure screening also with a non-polynomial $\log(p) = o(n^{-\kappa} n_{(1)}^{\gamma}) = o(\widehat{\pi}_{(1)}^{\kappa} n_{(1)}^{\gamma-\kappa})$, where $\widehat{\pi}_{(1)} = \frac{n_{(1)}}{n}$ is the smallest class proportion. Recall that, when Y is continuous, $\widehat{\pi}_{(1)} \approx \frac{1}{L}$ since we create slices containing approximately equal numbers of observations (see Section 2.2). Finally, we note that, similar to other model-free approaches such as DC-SIS (Li, Zhong, and Zhu 2012), CIS guarantees sure screening under much more generic conditions compared to SIS (Fan and Lv 2008)—in particular, CIS does not require a linear regression function for Y onto $\mathbf{X}$.

3. Simulation Study

In this section, we present simulation results on the performance of CIS in comparison to those of SIS (Fan and Lv 2008), HOLP (Wang and Leng 2016), SIRS (Zhu et al. 2011), DC-SIS (Li, Zhong, and Zhu 2012), and MDC-SIS (Shao and Zhang 2014). Some of the simulation scenarios are adapted from Zhu et al. (2011), Cui, Li, and Zhong (2015), and Chen, Fan, and Li (2018).

3.1. Summary Statistics to Assess Screening Performance

Because screening procedures are used as a preliminary step, followed by modeling and fitting efforts in which predictors are further assessed, their main priority is *sensitivity*; as we separate active from inactive predictors, we want to *minimize false negatives*; that is, cases in which $j \in \mathcal{A}$ but $j \notin \widehat{\mathcal{A}}$. Thus, to measure performance, we consider two summary statistics commonly used in the literature:

(a) For each simulated dataset, we compute the maximum rank achieved by true predictors $X_j, j \in \mathcal{A}$, or equivalently, the minimum rank in $\widehat{\mathcal{A}}$ required to ensure $\mathcal{A} \subseteq \widehat{\mathcal{A}}$. Following Zhu et al. (2011), we denote this by $\mathcal{R}$ and call it the *ranking measure*. $\mathcal{R}$ close to $s = |\mathcal{A}|$ is evidence of *ranking consistency*, another important property for feature screening procedures (see Zhu et al. 2011). In the result tables below, we present the median (median absolute deviation in parentheses) of R over $N = 1000$ simulated datasets corresponding to each simulation scenario.

(b) For each simulation scenario, fixing $d = |\widehat{\mathcal{A}}_d| = k \cdot (n/log(n))$ ($k = 1$, 2, or 3; see Section 2.5), we compute the proportion of simulated datasets (out of N) in which $\mathcal{A} \subseteq \widehat{\mathcal{A}}_d$. Following Shao and Zhang (2014), we denote this by $\mathcal{P}_a$. $\mathcal{P}_a$ close to 1 is evidence of *sure screening* for a procedure. To further investigate which among the active predictors are easier/harder to retain for a procedure, we also consider predictor-specific inclusion proportions denoted by $\mathcal{P}_j$, $j \in \mathcal{A}$. We call $\mathcal{P}_a$ and the $\mathcal{P}_j$'s as *inclusion measures*.

3.2. Simulation Scenarios

We create simulation scenarios based on the elements described below.

1. Sample size, number of predictors, and number of active predictors (n, p, s). We use $p = 2000$, $n = 200$ and 600, and $s = |\mathcal{A}|$ varying between 3 and 40 (this controls the *sparsity level*; the smaller the s, the sparser the problem). As the only exception, for Model(3)(d) we use $n = 117$ (see below).
2. Nature of the predictors. We simulate the p entries in the covariate vector $\mathbf{X}$ with different schemes. We start by drawing from a p-variate Gaussian $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_{\mathbf{X}})$ (see below for covariance specifications) and: (i) we keep the vector as drawn, to have p continuous predictors (Models(1), (2), (3)(a), (4)–(5)); (ii) we replace 50% of the entries in $\mathbf{X}$ with independently drawn binary predictors (Model(3)(b)); (iii) we replace 50% of the entries in $\mathbf{X}$ with independently drawn "perturbed" continuous predictors obtained from a Gaussian Mixture spiked with a very high variance component (Model(3)(c)). In addition, we consider (iv) predictors randomly selected from the ones in our real data application in Section 4 (Model(3)(d)).
3. Structure of the predictor covariance matrix. For (i)–(iii) above, the multivariate Gaussian $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_{\mathbf{X}})$ has four different covariance specifications: (i) $\Sigma_{\mathbf{X}}^{(I)} = \mathbb{I}_p$, the identity matrix (*Independent*); (ii) $\Sigma_{\mathbf{X}}^{(A)} = \{\sigma_{ij}\}$, $\sigma_{ij} = 0.8^{|i-j|}$, $i, j = 1, 2, \ldots, p$ (*Autoregressive*); (iii) $\Sigma_{\mathbf{X}}^{(B)} = \{\sigma_{ij}\}$, $\sigma_{ii} = 1$; $\sigma_{ij} = 0.4$ for $i \ne j$, i, j both $\in \mathcal{A}$ or both $\in \mathcal{I}$; and $\sigma_{ij} = 0.1$ for $i \in \mathcal{A}, j \in \mathcal{I}$ or $i \in \mathcal{I}, j \in \mathcal{A}$, $i, j = 1, 2, \ldots, p$ (*Block-structure*); and (iv) $\Sigma_{\mathbf{X}}^{(C)} = \{\sigma_{ij}\}$, $\sigma_{ii} = 1$; $\sigma_{ij} = 0.2$ for $i \ne j$, $i, j = 1, 2, \ldots, p$ (*Compound-symmetric*).
4. Response generating process. We generate a continuous $Y \in \mathbb{R}$ using single- or multi-index models. These comprise $m = 1, 2$, or 3 indexes (i.e., linear combinations of the predictors $X_j, j \in \mathcal{A}$) acting linearly or nonlinearly on the mean or the variance of $Y|\mathbf{X}$ (Models (1)–(5)).
5. Nature of the error. We always use additive errors and consider: (i) two homoscedastic cases, namely a Gaussian error $\epsilon_1 \sim N_1(0, 1)$ and a mixture of Gaussian errors $\epsilon_2 \sim 0.5 \times N_1(0, 1) + 0.5 \times N_1(0, 10^2)$; and (ii) a heteroscedastic case, namely a Gaussian error $\epsilon_3 \sim N_1(0, g^2(\sigma, \boldsymbol{\beta}^T\mathbf{X}))$.

6. Signal-to-noise ratio (SNR). We define SNR $= \frac{\text{var}(E(Y|\mathbf{X}))}{E(\text{var}(Y|\mathbf{X}))}$. When we use ϵ_1 or ϵ_2, all signal is contained in the mean $E(Y|\mathbf{X})$; $\text{var}(Y|\mathbf{X})$ does not depend on $\mathbf{X}$. In these cases, SNR has the standard definition. When we use ϵ_3, $\text{var}(Y|\mathbf{X}) = g^2(\sigma, \boldsymbol{\beta}^T\mathbf{X})$ itself contains "signal"; SNR benchmarks the signal in the mean to that in the variance. For the homoscedastic cases, we vary a scalar multiplier σ in the mean functions $E(Y|\mathbf{X})$, and for the heteroscedastic case, the σ in $g^2(\sigma, \boldsymbol{\beta}^T\mathbf{X})$, as to obtain SNRs ranging between 0.05 and 20 (see below).

In more detail, for the response generating process, we consider five models:

Model(1): Variation of Example 1 in Zhu et al. (2011).

Linear, single-index with homoscedastic additive error: $Y = \sigma \cdot (\boldsymbol{\beta}_1^T\mathbf{X}) + \epsilon_1$ with $\boldsymbol{\beta}_1 = (1, 0.8, 0.6, 0.4, 0.2, 0, \ldots, 0)^T$. Here $m = 1$ and $\mathcal{A} = \{1, 2, 3, 4, 5\}$ with $s = 5$. We use $n = 200$; $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_\mathbf{X})$ with both $\Sigma_\mathbf{X}^{(A)}$ and $\Sigma_\mathbf{X}^{(B)}$; and σ ranging between 0.34 and 2.02 to give rise to SNRs in the range ≈ 0.8 ("low") to ≈ 20 ("high"). Given its underlying assumptions, we included HOLP (Wang and Leng 2016) in our comparisons only for this model, where its performance should be among the best.

Model(2): Variation of Example 3.b in Zhu et al. (2011).

Multi-index with homoscedastic additive error: $Y = \boldsymbol{\beta}_1^T\mathbf{X} + \exp(\boldsymbol{\beta}_2^T\mathbf{X}) + \epsilon_1$ with $\boldsymbol{\beta}_1 = (2 - U_1, \ldots, 2 - U_{s/2}, 0, \ldots, 0)^T$, $\boldsymbol{\beta}_2 = (0, \ldots, 0, 2 + U_{s/2+1}, \ldots, 2 + U_s, 0, \ldots, 0)^T$, and U_k's independently drawn from a uniform distribution on $[0, 1]$. Here $m = 2$ and $\mathcal{A} = \{1, \ldots, s\}$. We use $s = 4, 8, 16, 24, 32$, and 40; $n = 200$; and $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_\mathbf{X})$ with both $\Sigma_\mathbf{X}^{(A)}$ and $\Sigma_\mathbf{X}^{(B)}$.

Model(3): Variation of Example 3.1 in Chen, Fan, and Li (2018).

Multi-index with homoscedastic additive error: $Y = \sigma \cdot (X_1 + 0.75X_2^2 + 2.25\cos(X_5)) + \epsilon_1$. Here $m = 3$ and $\mathcal{A} = \{1, 2, 5\}$ with $s = 3$. For this model, we implement different specifications, namely: (a) *Continuous predictors.* We use $n = 200$ and 600; $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_\mathbf{X})$ with both $\Sigma_\mathbf{X}^{(I)}$ and $\Sigma_\mathbf{X}^{(C)}$ (the compound-symmetric covariance should hinder screening, since the covariates possess sizable and equal correlations within and between $\mathcal{A}$ and $\mathcal{I}$); and $\sigma = 0.50, 1.25, 2.50$, respectively, giving rise to SNR $\approx$ 0.8 ("low"), $\approx$ 5 ("moderate"), and $\approx$ 20 ("high"). (b) *Mix of continuous and binary predictors.* We use $n = 200$; $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_\mathbf{X})$ with both $\Sigma_\mathbf{X}^{(I)}$ and $\Sigma_\mathbf{X}^{(C)}$, in which 50% of the entries (X_1 and a random selection from $X_6 - X_{2000}$) is replaced with 0/1 entries drawn independently with success probabilities equal to the sample qth quantiles of the X_j being replaced, using q from 0.30 to 0.70; and $\sigma = 1.5$ and 2.1, respectively, giving rise to SNR $\approx$ 5 ("moderate") and $\approx$ 10 ("high"). (c) *Mix of continuous and "perturbed" predictors.* We use $n = 200$ and 600; $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_\mathbf{X})$ with both $\Sigma_\mathbf{X}^{(I)}$ and $\Sigma_\mathbf{X}^{(C)}$, in which 50% of the entries (X_1 and a random selection from $X_6 - X_{2000}$) is replaced with entries independently drawn from the univariate Gaussian mixture $0.95 \times N_1(0, 1) + 0.05 \times N_1(0, 10^2)$; and $\sigma = 0.805$ and 1.14, respectively, giving rise to SNR $\approx$ 5 ("moderate") and $\approx$ 10 ("high"). (d) *"Realistic" predictors.* For each simulation repetitions, we randomly sub-sample $p = 2000$ predictors from the set of 18,941 gene/probe ID expressions in our transcriptomic application (see Section 4). We use the sample size $n = 117$ of that application, and $\sigma = 1.25$ and 1.77, respectively, giving rise to SNR $\approx$ 5 ("moderate") and $\approx$ 10 ("high"). Before generating the response Y, we standardize all predictors marginally to have mean 0 and variance 1. Since the sample size is smaller compared to the other simulation scenarios, we only use CIS with a small number of slices ($L = 2, 3$, and 5). Finally, to account for the complexity of this "realistic" predictor data, here we use the hard thresholds corresponding to $n = 600$, that is, $d = 93$ ($n/\log(n)$), 187 ($2n/\log(n)$), and 281 ($3n/\log(n)$).

Model(4)

Same as Model(3)(a), but with a mixture of Gaussian errors ϵ_2: $Y = \sigma \cdot (X_1 + 0.75X_2^2 + 2.25\cos(X_5)) + \epsilon_2$. The mixture induces heavier tails, and thus increased variance, for error (and response) compared to previous models.

Model(5).

Multi-index with heteroscedastic error: $Y = X_1 + X_2^2 + \epsilon_3$ with $g(\sigma, \boldsymbol{\beta}^T\mathbf{X}) = \exp\{\sigma|X_{22}|\}$. Here $m = 3$ and $\mathcal{A} = \{1, 2, 22\}$ with $s = 3$. We use $n = 200$ and 600; $\mathbf{X} \sim N_p(\mathbf{0}, \Sigma_\mathbf{X})$ with $\Sigma_\mathbf{X} = \Sigma_\mathbf{X}^{(A)}$; and σ in the range 0.23–1.37, giving rise to SNRs in the range 2–0.05. Recall that, instead of the standard definition, SNR here benchmarks the strength of the mean signal to that of the variance signal.

In addition to the above scenarios, all with a continuous response, we also investigate scenarios with a categorical response (see Model(6) in Section S3). For each of the scenarios described above, we simulate $N = 1000$ datasets to compute the performance summary statistics, and we assess the effect of the number of slices on CIS screening for varying L between 2 and 12.

3.3. Simulation Results

Due to space constraints, here we present results only for selected scenarios of Models (2) and (3)(a), and selected number of slices (L) used in CIS. Full results for all models with all scenarios and all choices of L are reported in Section S3.

Table 2 contains *ranking measures* ($\mathcal{R}$) for Model(2), summarizing performance under different predictor covariance structures and sparsity levels. Under the block covariance structure $\Sigma_\mathbf{X}^{(B)}$, CIS and SIRS outperform all other procedures for all values of s considered (s = 4–40 active predictors out of p = 2000). Under the autoregressive covariance structure $\Sigma_\mathbf{X}^{(A)}$, CIS and SIRS again outperform other procedures. However, as sparsity decreases ($s > 16$), CIS deteriorates faster than SIRS. A potential explanation is that under $\Sigma_\mathbf{X}^{(A)}$, as the number of active predictors s increases, more *inactive* predictors highly correlated with their adjacent active ones confound the CIS ranking. On the contrary, under $\Sigma_\mathbf{X}^{(B)}$, the level of correlation between active and inactive predictors is fixed at a relatively low 0.1. Notably, SIS, DC-SIS and MDC-SIS perform very poorly—except under $\Sigma_\mathbf{X}^{(A)}$ and very marked sparsity ($s = 4$).

Table 3 contains *inclusion measures* ($\mathcal{P}_a$ and $\mathcal{P}_j, j \in \mathcal{A} = \{1, 2, 3, 4\}$) for Model(2), again under both $\Sigma_\mathbf{X}^{(A)}$ and $\Sigma_\mathbf{X}^{(B)}$—but focusing on the $s = 4$ case. The excellent and comparable performance of CIS and SIRS is evident from these inclusion measures. Interestingly, the poorer performance of SIS, DC-SIS,

Table 2. Median (median absolute deviation) of the ranking measure $\mathcal{R}$ over $N = 1000$ simulated datasets. Model(2) with $n = 200$ and $p = 2000$. Sparsity (s) and the predictor covariance structure ($\Sigma_{\mathbf{X}}$) vary as indicated in the table. CIS results shown for $L = 5$ slices. Performance is better for values closer to s. $\Sigma_{\mathbf{X}}^{(A)}$ = autoregressive; $\Sigma_{\mathbf{X}}^{(B)}$ = block structure.

	SIS	SIRS	DC-SIS	MDC-SIS	CIS[5]	SIS	SIRS	DC-SIS	MDC-SIS	CIS[5]
	$s = 4$					$s = 24$				
$\Sigma_{\mathbf{X}}^{(A)}$	30 (25)	4 (0)	9 (5)	24 (20)	4 (0)	1852 (107)	29 (4)	1846 (111)	1860 (101)	84 (56)
$\Sigma_{\mathbf{X}}^{(B)}$	248 (231)	4 (0)	59 (55)	254 (242)	4 (0)	1427 (359)	24 (0)	1351 (366)	1513 (355)	24 (0)
	$s = 8$					$s = 32$				
$\Sigma_{\mathbf{X}}^{(A)}$	679 (502)	9 (1)	610 (454)	688 (519)	8 (0)	1918 (58)	52 (17)	1915 (62)	1917 (60)	382 (248)
$\Sigma_{\mathbf{X}}^{(B)}$	692 (462)	8 (0)	537 (403)	767 (502)	8 (0)	1600 (286)	32 (0)	1554 (287)	1644 (268)	33 (1)
	$s = 16$					$s = 40$				
$\Sigma_{\mathbf{X}}^{(A)}$	1694 (244)	18 (1)	1644 (261)	1692 (241)	19 (3)	1935 (45)	93 (43)	1931 (50)	1937 (46)	807 (403)
$\Sigma_{\mathbf{X}}^{(B)}$	1204 (456)	16 (0)	1130 (434)	1270 (450)	16 (0)	1695 (223)	40 (0)	1659 (241)	1723 (215)	41 (1)

Table 3. Inclusion measures $\mathcal{P}_a$ and $\mathcal{P}_j$, $j = 1, 2, 3, 4$, over $N = 1000$ simulated datasets, provided at thresholds $d = n/\log(n), 2n/\log(n), 3n/\log(n)$ (triplets in parentheses). Model(2) with $n = 200$, $p = 2000$, and $s = 4$. The predictor covariance structure ($\Sigma_{\mathbf{X}}$) varies as indicated in the table. CIS results shown for $L = 5$ slices. Values are multiplied by 10^3; closer to $1K = 1000$ indicate better performance. $\Sigma_{\mathbf{X}}^{(A)}$ = autoregressive; $\Sigma_{\mathbf{X}}^{(B)}$ = block structure.

$s = 4$	SIS		SIRS		DC-SIS		MDC-SIS		CIS[5]	
	$\Sigma_{\mathbf{X}}^{(A)}$	$\Sigma_{\mathbf{X}}^{(B)}$	$\Sigma_{\mathbf{X}}^{(A)}$	$\Sigma_{\mathbf{X}}^{(B)}$	$\Sigma_{\mathbf{X}}^{(A)}$	$\Sigma_{\mathbf{X}}^{(B)}$	$\Sigma_{\mathbf{X}}^{(A)}$	$\Sigma_{\mathbf{X}}^{(B)}$	$\Sigma_{\mathbf{X}}^{(A)}$	$\Sigma_{\mathbf{X}}^{(B)}$
$\mathcal{P}_1$	(572, 697, 730)	(411, 494, 538)	(1K, 1K, 1K)	(1K, 1K, 1K)	(750, 818, 865)	(593, 679, 733)	(594, 690, 734)	(426, 498, 538)	(1K, 1K, 1K)	(999, 999, 1K)
$\mathcal{P}_2$	(813, 888, 919)	(406, 488, 548)	(1K, 1K, 1K)	(1K, 1K, 1K)	(933, 964, 974)	(603, 678, 721)	(830, 895, 925)	(415, 498, 546)	(1K, 1K, 1K)	(997, 998, 998)
$\mathcal{P}_3$	(997, 1K, 1K)	(939, 967, 978)	(1K, 1K, 1K)	(1K, 1K, 1K)	(999, 1K, 1K)	(993, 995, 996)	(998, 1K, 1K)	(951, 972, 983)	(1K, 1K, 1K)	(1K, 1K, 1K)
$\mathcal{P}_4$	(996, 1K, 1K)	(927, 958, 969)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(984, 990, 993)	(997, 1K, 1K)	(943, 965, 977)	(1K, 1K, 1K)	(1K, 1K, 1K)
$\mathcal{P}_a$	(547, 658, 715)	(238, 315, 367)	(1K, 1K, 1K)	(1K, 1K, 1K)	(743, 812, 858)	(434, 538, 588)	(569, 669, 720)	(242, 315, 363)	(1K, 1K, 1K)	(996, 997, 998)

and MDC-SIS is driven by their inability to capture the active covariates X_1 and X_2 involved in the first index, that is, the linear component in $E(Y|\mathbf{X})$ (see Model(2)). While predictors acting linearly ought to be easy to identify, also with the Pearson correlation-based SIS, the exponential scale possibly renders the signals associated with X_3 and X_4 much stronger.

Table 4 contains *ranking measures* for Model(3)(a), summarizing performance under different predictor covariance structures, SNR levels, and sample sizes. Under both $\Sigma_{\mathbf{X}}^{(I)}$ and $\Sigma_{\mathbf{X}}^{(C)}$, the ranking performances of CIS, DC-SIS, and MDC-SIS beat those of SIS and SIRS for all SNRs and sample sizes. In particular, at moderate and high SNRs (5 and 20, respectively), CIS, DC-SIS, and MDC-SIS successfully rank the three active predictors as the top three. For higher sample size ($n = 600$), this ranking performance improves even at low SNR (0.8). Notably, although SIRS is a model-free procedure, it fails for Model(3)(a) in all scenarios—most likely due to the presence of $m = 3$ active indexes violating condition (C1) in Zhu et al. (2011).

Tables 5 ($n = 200$) and 6 ($n = 600$) containing the *inclusion measures* for Model(3)(a) support the sure screening property of CIS, under both $\Sigma_{\mathbf{X}}^{(I)}$ and $\Sigma_{\mathbf{X}}^{(C)}$, and low, moderate, and high SNRs. The excellent and comparable performance of CIS (except for low SNR *and* sample size), DC-SIS, and MDC-SIS is once again evident. The Pearson correlation-based SIS and, interestingly, also the model-free SIRS fail to capture X_2 and X_5—which are nonlinearly associated with Y. Notably, when $n = 200$, CIS performs better with $L = 3$ (CIS[3]) than with $L = 5$ (CIS[5]), likely due to more abundant observations available for in-slice calculations. In fact, for $n = 200$, SNR $= 5$ and 20, and $\Sigma_{\mathbf{X}}^{(C)}$ (the compound-symmetric structure that ought to hinder screening), the $\mathcal{P}_a$'s for CIS[3] are the highest.

The results described above represent only a small portion of the extensive simulation experiments we conducted (see Section 3.2). Below, we describe salient trends and observations based also on the additional results presented in Section S3. CIS exhibits promising performance in terms of sure screening as well as rank consistency under a wide range of scenarios. Overall, as expected intuitively, CIS performs better at larger SNRs and sample sizes, where it is less sensitive to the number of slices (L) used for a continuous Y. In Model(1) scenarios, CIS shows competitive sure screening and rank consistency performance compared to that of other screening procedures (Tables S1 and S2), especially for larger SNRs (5 and 20) and smaller L (3 and 5). In Model(2) scenarios, CIS does better than SIS, DC-SIS, and MDC-SIS; performs very good when sparsity is high and, while it tends to deteriorate at lower sparsity under one predictor covariance structure, it remains stable across sparsity levels for the other (Tables 2, 3, S3, and S4). In all the Model(3) scenarios ((a)–(d)) CIS does better than SIS and SIRS—which fail for reasons similar to those articulated above for Model(3)(a). In general, CIS has good performance at "moderate" and "high" SNRs, and its deterioration at lower SNR can be counteracted increasing the sample size and/or using a smaller number of slices to guarantee a sufficient number of observations per slice (Tables 4–6 and S5–S15). Moreover, the measures for inclusion ($\mathcal{P}_a$) and ranking ($\mathcal{R}$) provide empirical evidence for sure screening and ranking consistency of CIS, respectively. This is

Table 4. Median (median absolute deviation) of the ranking measure $\mathcal{R}$ over $N = 1000$ simulated datasets. Model(3)(a) with $p = 2000$ and $s = 3$. Sample size (n), SNR, and predictor covariance structure ($\Sigma_{\mathbf{X}}$) vary as indicated in the table. CIS results shown for $L = 5$ slices. Performance is better for values closer to $s = 3$. $\Sigma_{\mathbf{X}}^{(I)}$ = independent, $\Sigma_{\mathbf{X}}^{(C)}$ = compound-symmetric.

	SIS	SIRS	DC-SIS	MDC-SIS	CIS[5]	SIS	SIRS	DC-SIS	MDC-SIS	CIS[5]
	$n = 200$, SNR = 0.8					$n = 600$, SNR = 0.8				
$\Sigma_{\mathbf{X}}^{(I)}$	1279 (406)	1425 (365)	28 (19)	22 (14)	148 (127)	1333 (411)	1470 (358)	3 (0)	3 (0)	3 (0)
$\Sigma_{\mathbf{X}}^{(C)}$	1398 (379)	1451 (350)	80 (61)	65 (51)	190 (158)	1477 (352)	1474 (338)	4 (1)	4 (1)	3 (0)
	$n = 200$, SNR = 5					$n = 600$, SNR = 5				
$\Sigma_{\mathbf{X}}^{(I)}$	1230 (460)	1423 (375)	3 (0)	3 (0)	4 (1)	1183 (473)	1513 (318)	3 (0)	3 (0)	3 (0)
$\Sigma_{\mathbf{X}}^{(C)}$	1394 (382)	1485 (337)	10 (7)	11 (7)	5 (2)	1493 (372)	1504 (321)	3 (0)	3 (0)	3 (0)
	$n = 200$, SNR = 20					$n = 600$, SNR = 20				
$\Sigma_{\mathbf{X}}^{(I)}$	1231 (465)	1414 (398)	3 (0)	3 (0)	3 (0)	1246 (439)	1504 (335)	3 (0)	3 (0)	3 (0)
$\Sigma_{\mathbf{X}}^{(C)}$	1449 (379)	1562 (294)	6 (3)	7 (4)	3 (0)	1561 (346)	1522 (321)	3 (0)	3 (0)	3 (0)

Table 5. Inclusion measures $\mathcal{P}_a$ and $\mathcal{P}_j, j = 1, 2, 5$ over $N = 1000$ simulated datasets, provided at thresholds $d = n/\log(n), 2n/\log(n), 3n/\log(n)$ (triplets in parentheses). Model(3)(a) with $p = 2000$, $s = 3$, and $n = 200$. SNR and predictor covariance structure ($\Sigma_{\mathbf{X}}$) vary as indicated in the table. CIS results shown for $L = 3$ and 5 slices. Values are multiplied by 10^3; closer to $1K = 1000$ indicate better performance. $\Sigma_{\mathbf{X}}^{(I)}$ = independent, $\Sigma_{\mathbf{X}}^{(C)}$ = compound-symmetric.

		SNR ≈ 0.8		SNR ≈ 5		SNR ≈ 20	
$n = 200$		$\Sigma_{\mathbf{X}}^{(I)}$	$\Sigma_{\mathbf{X}}^{(C)}$	$\Sigma_{\mathbf{X}}^{(I)}$	$\Sigma_{\mathbf{X}}^{(C)}$	$\Sigma_{\mathbf{X}}^{(I)}$	$\Sigma_{\mathbf{X}}^{(C)}$
SIS	$\mathcal{P}_1$	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(60, 93, 118)	(49, 76, 104)	(101, 142, 185)	(62, 97, 132)	(111, 160, 207)	(73, 108, 144)
	$\mathcal{P}_5$	(46, 76, 109)	(22, 46, 72)	(60, 100, 132)	(44, 66, 83)	(72, 106, 138)	(49, 84, 107)
	$\mathcal{P}_a$	(3, 9, 11)	(0, 1, 3)	(6, 18, 29)	(2, 5, 12)	(3, 13, 25)	(2, 6, 10)
SIRS	$\mathcal{P}_1$	(999, 1K, 1K)	(999, 999, 999)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(27, 52, 89)	(23, 44, 64)	(36, 58, 86)	(18, 33, 50)	(52, 91, 121)	(27, 45, 63)
	$\mathcal{P}_5$	(36, 71, 94)	(19, 39, 62)	(39, 62, 89)	(25, 46, 57)	(41, 75, 93)	(18, 39, 56)
	$\mathcal{P}_a$	(0, 1, 7)	(1, 2, 3)	(2, 4, 7)	(0, 2, 3)	(0, 6, 11)	(0, 2, 2)
DC-SIS	$\mathcal{P}_1$	(999, 1K, 1K)	(999, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(748, 875, 915)	(489, 642, 734)	(997, 1K, 1K)	(840, 911, 948)	(999, 999, 1K)	(902, 949, 966)
	$\mathcal{P}_5$	(818, 923, 954)	(582, 735, 812)	(998, 1K, 1K)	(932, 969, 980)	(1K, 1K, 1K)	(974, 992, 998)
	$\mathcal{P}_a$	(601, 804, 871)	(314, 484, 610)	(995, 1K, 1K)	(788, 886, 929)	(999, 999, 1K)	(879, 941, 964)
MDC-SIS	$\mathcal{P}_1$	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(810, 907, 946)	(530, 688, 766)	(998, 1K, 1K)	(862, 933, 955)	(999, 1K, 1K)	(914, 960, 977)
	$\mathcal{P}_5$	(841, 940, 967)	(582, 750, 816)	(1K, 1K, 1K)	(901, 952, 973)	(1K, 1K, 1K)	(949, 980, 986)
	$\mathcal{P}_a$	(671, 849, 914)	(331, 536, 639)	(998, 1K, 1K)	(780, 887, 929)	(999, 1K, 1K)	(865, 940, 963)
CIS[3]	$\mathcal{P}_1$	(639, 731, 779)	(591, 683, 740)	(996, 998, 998)	(990, 993, 994)	(999, 999, 999)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(680, 749, 788)	(610, 701, 745)	(942, 964, 973)	(929, 950, 961)	(960, 980, 985)	(944, 962, 973)
	$\mathcal{P}_5$	(787, 843, 872)	(735, 799, 845)	(984, 989, 993)	(969, 979, 984)	(997, 999, 1K)	(993, 998, 999)
	$\mathcal{P}_a$	(335, 455, 532)	(245, 369, 454)	(923, 952, 965)	(891, 923, 939)	(956, 978, 984)	(937, 960, 972)
CIS[5]	$\mathcal{P}_1$	(572, 674, 736)	(538, 635, 701)	(989, 994, 997)	(984, 992, 996)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(628, 726, 766)	(551, 639, 697)	(910, 943, 956)	(899, 941, 957)	(947, 965, 973)	(927, 948, 956)
	$\mathcal{P}_5$	(723, 792, 825)	(673, 761, 802)	(967, 977, 980)	(958, 972, 980)	(993, 995, 998)	(987, 991, 996)
	$\mathcal{P}_a$	(235, 372, 443)	(184, 298, 380)	(868, 915, 933)	(843, 906, 933)	(940, 960, 971)	(914, 939, 952)

true when all predictors are continuous (Model(3)(a); Tables 4–6, S5–S7), but also in cases where continuous predictors are mixed with categorical predictors (Model(3)(b); Tables S8 and S9), or with "perturbed" continuous predictors (Model(3)(c); Tables S10–S13), and when predictors are sub-sampled from real data (Model(3)(d); Tables S14 and S15). Notably, in the latter "realistic" scenario, which carries substantial collinearities (see Section 4 and Figure S1), CIS with $L = 3$ slices (CIS[3]) performs the best, beating also the otherwise strongest competitor DC-SIS. CIS performs very well also with the heavier-tailed error of Model(4) (Tables S16–S18; results are similar to those for Model(3)(a)). When $n = 200$, almost all $\mathcal{P}_a$'s for CIS[3] beat those for DC-SIS and MDC-SIS under $\Sigma_{\mathbf{X}}^{(C)}$ and, with a larger sample size ($n = 600$), CIS performs competitively with these methods in all respects. SIS and SIRS fail in all Model(4) scenarios for reasons similar to those discussed for Model(3)(a). In Model(5) scenarios, where the error is heteroscedastic, CIS and DC-SIS are the best overall performers (Tables S19–S21). When $n = 200$, ranking performance of CIS with $L = 5$ slices (CIS[5]) is better than with smaller or higher L, likely due to the added complexity of capturing signals in the variance (as opposed to the mean). When $n = 600$, once again CIS performs well, along with DC-SIS, across all $L = 3$–12 and SNR levels less than 1. Notably, MDC-SIS always fails to capture the active predictor X_{22} present in the variance component of the model (Table S19). This is because its marginal utility is designed to detect predictors contributing to the conditional mean of the response (Shao and Zhang 2014). Finally, results for Model(6) scenarios demonstrate that CIS performs quite well also in problems with categorical responses (Tables S22–S24).

Table 6. Same as Table 5 for Model(3)(a), with sample size $n = 600$. Model(3)(a) with $p = 2000$, $s = 3$, and $n = 200$. SNR and predictor covariance structure ($\Sigma_{\mathbf{X}}$) vary as indicated in the table. CIS results shown for $L = 3$ and 5 slices. Values are multiplied by 10^3; closer to $1K = 1000$ indicate better performance. $\Sigma_{\mathbf{X}}^{(I)}$ = Independent, $\Sigma_{\mathbf{X}}^{(C)}$ = Compound-symmetric.

		SNR ≈ 0.8		SNR ≈ 5		SNR ≈ 20	
$n = 600$		$\Sigma_{\mathbf{X}}^{(I)}$	$\Sigma_{\mathbf{X}}^{(C)}$	$\Sigma_{\mathbf{X}}^{(I)}$	$\Sigma_{\mathbf{X}}^{(C)}$	$\Sigma_{\mathbf{X}}^{(I)}$	$\Sigma_{\mathbf{X}}^{(C)}$
SIS	$\mathcal{P}_1$	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(103, 165, 231)	(86, 142, 193)	(168, 249, 307)	(107, 170, 213)	(170, 234, 290)	(125, 187, 238)
	$\mathcal{P}_5$	(66, 121, 184)	(72, 132, 187)	(122, 187, 241)	(89, 142, 188)	(115, 174, 237)	(78, 132, 182)
	$\mathcal{P}_a$	(2, 11, 41)	(7, 16, 30)	(22, 53, 81)	(7, 19, 36)	(18, 44, 68)	(10, 22, 40)
SIRS	$\mathcal{P}_1$	(885, 948, 975)	(875, 942, 967)	(997, 999, 1K)	(998, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(22, 52, 86)	(22, 58, 95)	(15, 44, 74)	(21, 45, 83)	(6, 28, 57)	(16, 48, 78)
	$\mathcal{P}_5$	(77, 138, 195)	(82, 130, 185)	(123, 181, 230)	(90, 157, 209)	(106, 177, 226)	(85, 136, 185)
	$\mathcal{P}_a$	(0, 4, 14)	(2, 5, 15)	(1, 7, 19)	(2, 4, 13)	(2, 7, 14)	(0, 3, 9)
DC-SIS	$\mathcal{P}_1$	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(1K, 1K, 1K)	(990, 997, 999)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_5$	(1K, 1K, 1K)	(995, 999, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_a$	(1K, 1K, 1K)	(985, 996, 999)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
MDC-SIS	$\mathcal{P}_1$	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(1K, 1K, 1K)	(991, 998, 998)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_5$	(1K, 1K, 1K)	(994, 999, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_a$	(1K, 1K, 1K)	(985, 997, 998)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
CIS[3]	$\mathcal{P}_1$	(994, 999, 999)	(991, 996, 996)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(987, 992, 995)	(979, 993, 997)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_5$	(1K, 1K, 1K)	(998, 999, 999)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_a$	(981, 991, 994)	(968, 988, 992)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
CIS[5]	$\mathcal{P}_1$	(989, 992, 993)	(986, 993, 996)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_2$	(984, 992, 995)	(976, 982, 988)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_5$	(998, 998, 999)	(997, 998, 999)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)
	$\mathcal{P}_a$	(971, 982, 987)	(959, 973, 983)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)	(1K, 1K, 1K)

4. Application to Transcriptomic Data

In this section, we analyze a transcriptomic dataset (Affymetrix GeneChip Rat Genome 230 2.0 Array Data; Scheetz et al. 2006) already used in feature screening and variable selection literature (Huang, Horowitz, and Wei 2010; Fan, Feng, and Song 2011; Wang, Wu, and Li 2012; Shao and Zhang 2014; Wang and Leng 2016). The expression Quantitative Trait Loci (eQTL) experiment in Scheetz et al. (2006) used gene transcription measurements from 120 12-week-old male F2 Norway rats (*Rattus norvegicus*) to better understand gene regulation in the mammalian eye, with potential relevance to the study of human eye disease. The dataset is publicly available at the NCBI Gene Expression Omnibus with accession number GSE5680.

Following Shao and Zhang (2014), we preprocess the data taking log transformations and eliminating all genes (more accurately, probe IDs) that do not show sufficient variation across rats—which leaves us with 18,976 genes. As in prior screening exercises conducted on this dataset, we consider as response the transcription of TRIM32 (probe ID: 1389163_at), a gene with a causal association with Bardet–Biedl syndrome which affects multiple human systems (Chiang et al. 2006). As a further preprocessing step, we identify outliers detected by both the built-in R statistical software function *boxplot()* and the "thrice median absolute deviation rule" (Barghash, Arslan, and Helms 2016) (see for instance Figure S2). We eliminate 34 genes that contain more than 12 outliers (10% of the total number of rats). Also, we omit three outliers detected for the response. Thus, we eventually work with a dataset comprising transcription levels for $p = 18{,}941$ genes (the predictors) and transcription levels for TRIM32 (the response) measured on $n = 117$ rats (the observations). On this dataset, we apply our CIS with $L = 3$ slices (CIS[3]), as well as SIS, HOLP, SIRS, DC-SIS, and MDC-SIS screening procedures, and *GAMSEL* (Chouldechova and Hastie 2015)—a generalized additive model selection procedure.

First, for each gene, we exclude outliers (12 or fewer values) and compute marginal utilities for all screening procedures considered. Next, we focus on the top ranked $d = 10$ genes, which differ substantially across procedures (Table S26)—likely due to linear associations among predictors (the absolute values of pair-wise Pearson correlations range between 0 and 0.9812; first quartile 0.0847, median 0.1796, third quartile 0.3053) and/or other complexities of the problem (e.g., level of sparsity, strength and nature of the signals). Notably though, two of the top 10 CIS[3] genes (ranks 1 and 6) are also within the top genes reported in Fan, Feng, and Song (2011). Also, the genes ranked 6 and 9 by CIS[3] are placed in the top 10 by all other procedures (except HOLP). DC-SIS and SIRS also include the gene ranked 5 by CIS[3] in their top 10. Interestingly, none of the top 10 genes for HOLP overlap with the top 10 of any other procedure considered.

Figure 1 illustrates the marginal associations between the transcription of TRIM32 (response) and those of the top 10 CIS[3] genes (predictors). The panels for the genes ranked 5, 8, and 10 clearly show nonlinear relationships, supporting the notion that our CIN, unlike the marginal utility used by SIS, is a general measure of association. Preliminary queries indicate that the top 10 CIS[3] genes do indeed have biological significance. Most of them are *conserved* in other mammalian and

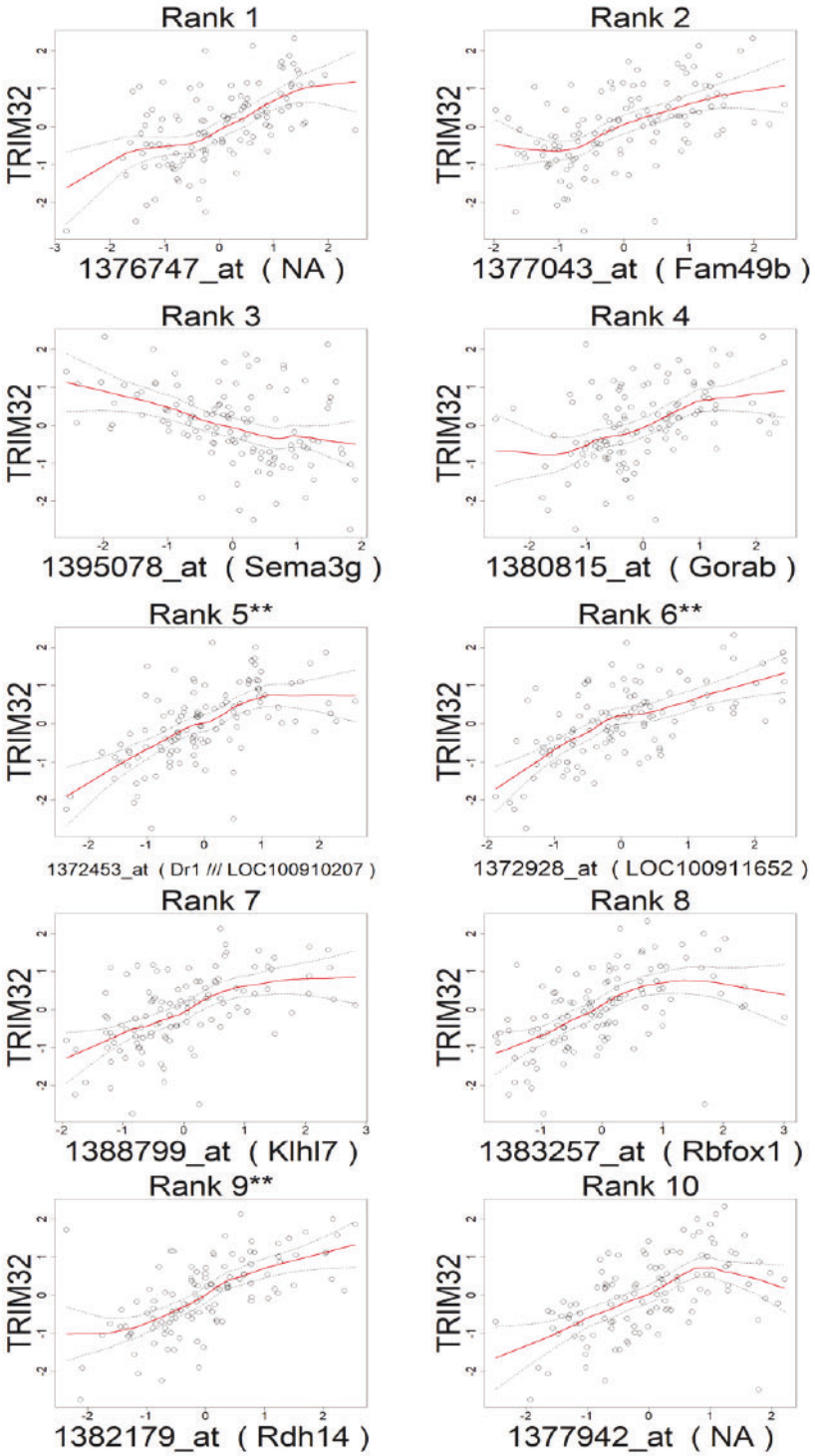

Figure 1. Scatterplots of transcription levels of TRIM32 against each of the top $d = 10$ genes identified by CIS[3]. Solid lines are LOESS smooths; dashed lines are 2-SD prediction bands. Genes marked as ** are also among the top $d = 10$ of other screening procedures (Rank 5, DC-SIS and SIRS; Ranks 6 and 9, all but HOLP).

vertebrate species including human, chimpanzee, Rhesus monkey, dog, cow, mouse, chicken, zebrafish, and frog—suggesting that they fulfill critical functions in the genome. Klhl7 (the gene ranked 7), is involved in the eye disease *Retinitis pigmentosa* (RP) in Norway rats; see the Rat Genome Database (RGD), ID *1305564*. According to the Online Mendelian Inheritance in Man (OMIM) database, its human ortholog KLHL7 also plays a role in human RP (OMIM ID *611119*; Friedman et al. 2009; Wen et al. 2011). In addition, according to the Mouse Genome Informatics (MGI) database, during wild-type mice development Klhl7 is expressed in the retina (MGI ID *1196453*), Rbfox1 (the gene ranked 8) in the retina ganglion cell layer (MGI ID *1926224*), and Dr1 (the gene ranked 5) in the retinal inner and outer layers (MGI ID *1100515*). Mutations in Fam49b (the gene ranked 2) are involved in abnormal retinal morphology in mice (MGI ID *1923520*). Finally, not directly related to the eye, Gorab

(the gene ranked 4) plays a role in gerodermia osteodysplastica, osteoporosis and skin abnormalities in Norway rats (RGD ID: *1564990*).

When we increase d from 10 to 5000, the overlaps across the top genes identified by various screening procedures increase (e.g., the top 5000 CIS[3] and DC-SIS genes have ≈72% overlap; see Table S26). Following Wang and Leng (2016), we consider the top 5000 genes produced by each screening procedure for subsequent modeling. We marginally standardize each set of 5000 top-ranked predictors, as well as the response, to zero mean and unit variance and employ GAMSEL (Chouldechova and Hastie 2015), a penalized likelihood approach for fitting sparse generalized additive models in high dimension, using the CRAN package *gamsel* (Chouldechova, Hastie, and Spinu 2018). We also apply GAMSEL directly on all (standardized) $p = 18{,}941$ predictors without any screening. We tune the *overall penalty parameter* ($\lambda \geq 0$) by 10-fold cross-validation, fixing the folds across runs for reproducibility and selecting the largest λ with cross-validation error within 1 standard error of the minimum. We set the *penalty mixing parameter* ($0 \leq \gamma \leq 1$; values <0.5 penalize the linear fit less than the nonlinear fit) to $\gamma = 0.6$ and the *degrees* (the maximum number of spline basis functions to use) to 5 for each predictor. All other parameters are left at their default values.

Table 7 shows that CIS[3] leads to the highest deviance explained (81.64%), followed by SIS (81.11%). Notably, GAMSEL applied to all $p = 18{,}941$ predictors leads to the lowest deviance explained. To provide a benchmark, we create a "null" distribution as follows: we select $d = 5000$ genes at random 1000 times, each time fitting GAMSEL and producing the corresponding deviance explained. The density plot in Figure 2 shows that the deviance explained with 5000 CIS[3]-screened genes is significantly larger than those expected when randomly selecting 5000 genes. In contrast, the deviance explained with 5000 HOLP-screened genes is not. Nor is the deviance explained with GAMSEL applied to all genes.

Finally, we evaluate the out-of-sample performance of GAMSEL fits on the $d = 5000$ top ranked genes produced by each screening procedure, as well as on all $p = 18941$ genes, and on a random selection of 5000 genes for benchmarking. We produce 200 90%–10% training-validation random splits of the $n = 117$ observations. We run GAMSEL fits on the training sets, and compute prediction errors on the corresponding validation sets. Figures S3(a)-(c) display boxplots of, respectively, the training-set deviance explained (in %), the number of nonzero training-set coefficient estimates obtained with 10-fold cross-validated λ's, and the validation-set root mean squared prediction error (RMSPE). On the training sets, HOLP and GAMSEL applied to all genes have similar median deviance explained (≈50%) and number of nonzero coefficients (≈15), which are comparable to those achieved with a random selection of 5000 genes. All other screening procedures lead to better fits (median deviance explained in the range ≈65%–70%) and larger gene sets (median number of nonzero coefficient estimates ≈25). Perhaps not surprisingly, given the small sizes of both training and validation sets (106 and 11, respectively) compared to dimensionality ($d = 5000$ for GAMSEL following screens, and $p = 18{,}941$ for GAMSEL directly applied to all genes), for *all* procedures the RMSPE's computed on the validation sets are on par with those obtained with a random selection of 5000 genes.

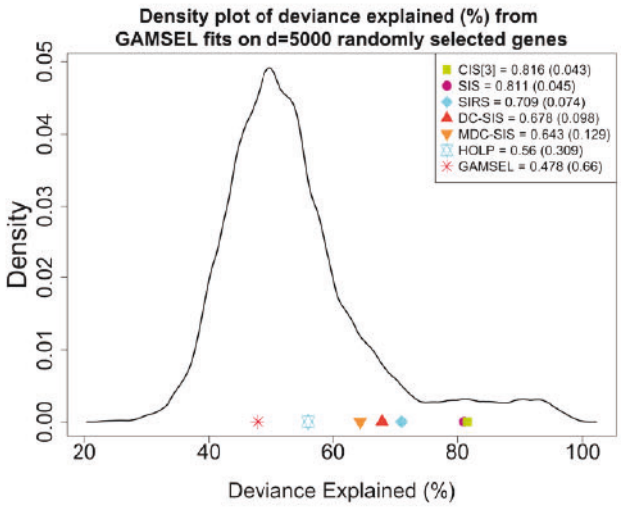

Figure 2. Density plot of deviance explained (%) from 1000 GAMSEL fits, each using $d = 5000$ randomly selected genes/probe IDs. Symbols mark performance achieved with those identified by CIS[3], SIS, SIRS, DC-SIS, MDC-SIS, and HOLP screens or directly applied to all $p = 18{,}941$ ("GAMSEL"); empirical p-values in parentheses in the legends.

5. Concluding Remarks

In this article, we proposed CIS—a model-free feature screening procedure to reduce the predictor dimension in ultrahigh-dimensional supervised problems prior to the use of other statistical techniques for feature selection, dimension reduction, and regression or classification modeling.

CIS is built upon the CIN, a novel marginal utility which is essentially the univariate version of the covariate information matrix (Yao et al. 2019) and has an appealing interpretation in terms of the traditional Fisher information in statistics. It is applicable to any type of response (features)—continuous, discrete, or categorical—with continuous features (response), has a reasonable computational burden, and possesses the important *sure screening property*.

Our simulation results demonstrate that CIS is competitive with, and in some cases superior to, popular feature screening procedures such as SIS, HOLP, SIRS, DC-SIS, and MDC-SIS. CIS successfully identifies active covariates at all levels of sparsity, with both continuous and categorical responses as well

Table 7. Deviance explained (%) and number of nonzero coefficient estimates selected by 10-fold cross-validated λ ("lambda.1se") from GAMSEL fits on the top $d = 5000$ genes/probe IDs ranked by different screens or directly applied to all $p = 18{,}941$ ("GAMSEL").

	SIS	HOLP	SIRS	DC-SIS	MDC-SIS	CIS[3]	GAMSEL
Deviance explained (%)	81.11	56.04	70.88	67.83	64.32	81.64	47.76
# of nonzero coefficients	48	18	32	30	28	45	14

as categorical, "perturbed," and "realistic" predictors. As to be expected, it outperforms SIS in the presence of nonlinear signals but, notably, it also outperforms DC-SIS and MDC-SIS in less sparse settings (higher number of active covariates). Moreover, it outperforms SIRS when active covariates affect the response through more than two linear combinations (i.e., indexes in multi-index models). Importantly, in addition to sure screening, our simulation results provide empirical evidence that CIS possesses the *ranking consistency property*.

Like most procedures, the performance of CIS improves with higher sample sizes and signal-to-noise ratios. The former is particularly relevant because CIS calculations require a reasonable number of observations per slice. Our general suggestion is to employ a relatively small number of slices (say, $L = 3$–8). Notably though, L ceases to affect CIS performance when the sample size is sufficiently large. Switching to the case of discrete or categorical responses, where L represents the number of distinct Y values, we note that this (similar to the number of active predictors and that of predictors overall) can increase with n. We considered a finite L to theoretically establish the sure screening property for CIS, but the proof could potentially be generalized to a *diverging L*.

While the sure screening property addresses false negatives concerns, screens can retain *false positives* in cases where correlations between inactive and active covariates produce spurious association with the response (Fan and Lv 2008). One way to mitigate this issue is to use *iteration*. For example, an iterative model-based screening procedure can be found in Fan, Samworth, and Wu (2009). Iterative model-free screening procedures also exist, and are often based on the notion of *predictor residual matrix*. This was first introduced for *iterative SIRS* in Zhu et al. (2011) and later used for *iterative DC-SIS* (Zhong and Zhu 2015). An *iterative CIS* could be developed as well. Of course, iteration increases computational burden. In practice, an evaluation of the strength and structure of the associations among covariates can help gauge whether such burden is justified as a way to reduce potential false positives. Further discussion on iteration can be found in Fan, Samworth, and Wu (2009). The interested reader can also refer to *univariate penalization screening* (UPS; Ji and Jin 2012), *covariance assisted screening and estimation* (CASE; Ke, Jin, and Fan 2014), and *graphlet screening* (Jin, Zhang, and Zhang 2014), among others, for ideas on two-stage "screen and clean" procedures to tackle potential false positives.

We foresee several additional avenues for future work. One is *combining different screening approaches*. For instance, consider a composite marginal utility of the form $\omega(\tau) = \tau\omega_{S(1)} + (1-\tau)\omega_{S(2)}$, where $S(1)$ and $S(2)$ indicate two different screens and $\tau \in [0,1]$ is a weighing parameter. $\omega(\tau)$, especially with an appropriate data-driven tuning of τ, could combine the strengths of different approaches. As another instance, consider the selection of the threshold d used for separating active and inactive covariates (both soft and hard thresholding rules are discussed in the literature; see Fan and Fan 2008; Zhu et al. 2011; Li, Zhong, and Zhu 2012; Shao and Zhang 2014). Let $c(d) = |\widehat{\mathcal{A}}_d^{S(1)} \bigcap \widehat{\mathcal{A}}_d^{S(2)}|$ be the cardinality of the intersection of the active sets estimated by two screens using d. By construction $c(d) \leq d$; a plot of $c(d)$ versus d, $d = 1, 2, \ldots$, can be used as a visual diagnostics to identify d^* where $c(d^*)$ comes very close to d^*, that is, a threshold that guarantees high congruence between screens. Both the composite marginal utility and the threshold diagnostic plot, of course, could potentially combine more than two screens.

Another interesting future avenue is investigating the performance of CIS in the *rare and weak* signal regimes often encountered in Genome Wide Association Studies (GWAS), and in cases where the assumption that zero low-order marginal correlations imply zero higher-order partial correlations (also known as the "faithfulness" condition; Genovese et al. 2012) is violated due to factors such as *signal cancellation* (Wasserman and Roeder 2009). Procedures such as the *covariance assisted screening and estimation* (Ke, Jin, and Fan 2014) and the *graphlet screening* (Jin, Zhang, and Zhang 2014) address these issues.

We mentioned (Section 1) and demonstrated via numerical examples (Sections 3.2 and S3) that CIS can also be used to screen discrete or categorical predictors—as long as the response is continuous. Another important avenue for future work is the extension of CIS to cases where both the response and the covariates are discrete or categorical, as well as to cases where the response is multivariate. Developments in the former (e.g., Huang, Li, and Wang 2014; Cui, Li, and Zhong 2015) and the latter (e.g., Zhu et al. 2011; Li, Zhong, and Zhu 2012; Shao and Zhang 2014) directions already exist in the feature screening literature.

Supplementary Materials and Codes

Proofs of theoretical results, full simulation results, details on the transcriptomic data application, and some relevant additional information are provided in an online Supplement. MATLAB (MATLAB 2020) and R (R Core Team 2020) source functions for the implementation of CIS and other feature screening procedures, codes for the numerical examples in the simulation study, and the analyses of the transcriptomic data are publicly available at the following link: bit.ly/CIS-Codes.

Acknowledgments

We thank Drs. Bharath Sriperumbudur, Amal Agarwal, and Mauricio Nascimento for helping with theoretical derivations; Dr. Weixin Yao for MATLAB codes to compute Covariate Information Matrices; Dr. Paolo Inglese for MATLAB code to compute distance correlations; and Drs. Xiaofeng Shao and Jingsi Zhang for R code to compute martingale difference correlations, the transcriptomic data, and R codes for its preprocessing. We also thank members of the Makova Lab at Penn State and Binglan (Victoria) Li for helping with the transcriptomic data application. Finally, we are grateful to the anonymous reviewers and the associate editor for crucial feedback that helped us greatly to improve our work.

Funding

F. Chiaromonte and D. Nandy were supported by NSF grant DMS-1407639. R. Li was supported by NSF grants DMS-1820702, DMS-1953196, and DMS-2015539, and NIH grants R01CA229542, R01ES019672, and R21CA226300.

References

Barghash, A., Arslan, T., and Helms, V. (2016), "Robust Detection of Outlier Samples and Genes in Expression Datasets," *Journal of Proteomics and Bioinformatics*, 9, 38–48. [1525]

Bickel, P. J., and Levina, E. (2008), "Regularized Estimation of Large Covariance Matrices," *The Annals of Statistics*, 36, 199–227. [1516]

Chen, Y., Chi, Y., and Goldsmith, A. J. (2015), "Exact and Stable Covariance Estimation From Quadratic Sampling via Convex Programming," *IEEE Transactions on Information Theory*, 61, 4034–4059. [1516]

Chen, Z., Fan, J., and Li, R. (2018), "Error Variance Estimation in Ultrahigh-Dimensional Additive Models," *Journal of the American Statistical Association*, 113, 315–327. [1521,1522]

Chiang, A. P., Beck, J. S., Yen, H.-J., Tayeh, M. K., Scheetz, T. E., Swiderski, R. E., Nishimura, D. Y., Braun, T. A., Kim, K.-Y. A., Huang, J., Elbedour, K., Carmi, R., Slusarski, D. C., Casavant, T. L., Stone, E. M., and Sheffield, V. C. (2006), "Homozygosity Mapping With SNP Arrays Identifies TRIM32, an E3 Ubiquitin Ligase, as a Bardet–Biedl Syndrome Gene (BBS11)," *Proceedings of the National Academy of Sciences of the United States of America*, 103, 6287–6292. [1525]

Chouldechova, A., and Hastie, T. (2015), "Generalized Additive Model Selection," arXiv no. 1506.03850. [1525,1527]

Chouldechova, A., Hastie, T., and Spinu, V. (2018), "gamsel: Fit Regularization Path for Generalized Additive Models," R Package Version 1(1). [1527]

Cook, R. D., and Weisberg, S. (1991), "Comment," *Journal of the American Statistical Association*, 86, 328–332. [1518]

Cui, H., Li, R., and Zhong, W. (2015), "Model-Free Feature Screening for Ultrahigh Dimensional Discriminant Analysis," *Journal of the American Statistical Association*, 110, 630–641. [1521,1528]

Fan, J., and Fan, Y. (2008), "High Dimensional Classification Using Features Annealed Independence Rules," *The Annals of Statistics*, 36, 2605. [1528]

Fan, J., Fan, Y., and Wu, Y. (2011), "High-Dimensional Classification," in *High-Dimensional Data Analysis*, eds by T. Cai and X. Shen, World Scientific, Singapore. pp. 3–37. [1516]

Fan, J., Feng, Y., and Song, R. (2011), "Nonparametric Independence Screening in Sparse Ultra-High-Dimensional Additive Models," *Journal of the American Statistical Association*, 106, 544–557. [1525]

Fan, J., and Li, R. (2001), "Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties," *Journal of the American Statistical Association*, 96, 1348–1360. [1516]

Fan, J., and Lv, J. (2008), "Sure Independence Screening for Ultrahigh Dimensional Feature Space," *Journal of the Royal Statistical Society*, Series B, 70, 849–911. [1516,1517,1518,1520,1521,1528]

——— (2010), "A Selective Overview of Variable Selection in High Dimensional Feature Space," *Statistica Sinica*, 20, 101–148. [1516]

Fan, J., Samworth, R., and Wu, Y. (2009), "Ultrahigh Dimensional Feature Selection: Beyond the Linear Model," *Journal of Machine Learning Research*, 10, 2013–2038. [1516,1528]

Friedman, J. S., Ray, J. W., Waseem, N., Johnson, K., Brooks, M. J., Hugosson, T., Breuer, D., Branham, K. E., Krauth, D. S., Bowne, S. J., and Sullivan, L. S. (2009), "Mutations in a BTB-Kelch Protein, KLHL7, Cause Autosomal-Dominant Retinitis Pigmentosa," *The American Journal of Human Genetics*, 84, 792–800. [1526]

Genovese, C. R., Jin, J., Wasserman, L., and Yao, Z. (2012), "A Comparison of the Lasso and Marginal Regression," *The Journal of Machine Learning Research*, 13, 2107–2143. [1528]

Huang, D., Li, R., and Wang, H. (2014), "Feature Screening for Ultrahigh Dimensional Categorical Data With Applications," *Journal of Business & Economic Statistics*, 32, 237–244. [1528]

Huang, J., Horowitz, J. L., and Wei, F. (2010), "Variable Selection in Nonparametric Additive Models," *The Annals of Statistics*, 38, 2282–2313. [1525]

Hui, G., and Lindsay, B. G. (2010), "Projection Pursuit via White Noise Matrices," *Sankhya B*, 72, 123–153. [1518]

Ji, P., and Jin, J. (2012), "UPS Delivers Optimal Phase Diagram in High-Dimensional Variable Selection," *The Annals of Statistics*, 40, 73–103. [1528]

Jin, J., Zhang, C.-H., and Zhang, Q. (2014), "Optimality of Graphlet Screening in High Dimensional Variable Selection," *The Journal of Machine Learning Research*, 15, 2723–2772. [1528]

Ke, T., Jin, J., and Fan, J. (2014), "Covariance Assisted Screening and Estimation," *The Annals of Statistics*, 42, 2202–2242. [1528]

Ritchie, D. M., Kernighan, B. W., and Lesk, M. E. (1988). The C programming language. Englewood Cliffs: Prentice Hall. [1519]

Ledoit, O., and Wolf, M. (2004), "A Well-Conditioned Estimator for Large-Dimensional Covariance Matrices," *Journal of Multivariate Analysis*, 88, 365–411. [1516]

Li, K.-C. (1991), "Sliced Inverse Regression for Dimension Reduction," *Journal of the American Statistical Association*, 86, 316–327. [1518]

Li, R., Zhong, W., and Zhu, L. (2012), "Feature Screening via Distance Correlation Learning," *Journal of the American Statistical Association*, 107, 1129–1139. [1517,1520,1521,1528]

Lindsay, B. G., and Yao, W. (2012), "Fisher Information Matrix: A Tool for Dimension Reduction, Projection Pursuit, Independent Component Analysis, and More," *Canadian Journal of Statistics*, 40, 712–730. [1518]

Liu, J., Li, R., and Wu, R. (2014), "Feature Selection for Varying Coefficient Models With Ultrahigh-Dimensional Covariates," *Journal of the American Statistical Association*, 109, 266–274. [1520,1521]

Liu, J., Zhong, W., and Li, R. (2015), "A Selective Overview of Feature Screening for Ultrahigh-Dimensional Data," *Science China Mathematics*, 58, 1–22. [1516]

MATLAB (2020), *MATLAB Version 9.9.0.1467703 (R2020b)*, Natick, MA: The MathWorks Inc. [1519,1528]

R Core Team (2020), *R: A Language and Environment for Statistical Computing*, Vienna, Austria: R Foundation for Statistical Computing. [1519,1528]

Schäfer, J., and Strimmer, K. (2005), "A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics," *Statistical Applications in Genetics and Molecular Biology*, 4, 1175–1189. [1516]

Scheetz, T. E., Kim, K.-Y. A., Swiderski, R. E., Philp, A. R., Braun, T. A., Knudtson, K. L., Dorrance, A. M., DiBona, G. F., Huang, J., Casavant, T. L., Sheffield, V. C., and Stone, E. M. (2006), "Regulation of Gene Expression in the Mammalian Eye and Its Relevance to Eye Disease," *Proceedings of the National Academy of Sciences of the United States of America*, 103, 14429–14434. [1517,1525]

Shao, X., and Zhang, J. (2014), "Martingale Difference Correlation and Its Use in High-Dimensional Variable Screening," *Journal of the American Statistical Association*, 109, 1302–1318. [1517,1520,1521,1524,1525,1528]

Silverman, B. W. (2018), *Density Estimation for Statistics and Data Analysis*, Abingdon: Routledge. [1519]

Székely, G. J., Rizzo, M. L., and Bakirov, N. K. (2007), "Measuring and Testing Dependence by Correlation of Distances," *The Annals of Statistics*, 35, 2769–2794. [1517]

Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society*, Series B, 58, 267–288. [1516]

Wang, H., and Xia, Y. (2008), "Sliced Regression for Dimension Reduction," *Journal of the American Statistical Association*, 103, 811–821. [1518,1519]

Wang, L., Wu, Y., and Li, R. (2012), "Quantile Regression for Analyzing Heterogeneity in Ultra-High Dimension," *Journal of the American Statistical Association*, 107, 214–222. [1525]

Wang, X., and Leng, C. (2016), "High Dimensional Ordinary Least Squares Projection for Screening Variables," *Journal of the Royal Statistical Society*, Series B, 78, 589–611. [1517,1521,1522,1525,1527]

Wasserman, L., and Roeder, K. (2009), "High Dimensional Variable Selection," *The Annals of Statistics*, 37, 2178–2201. [1528]

Wen, Y., Locke, K. G., Klein, M., Bowne, S. J., Sullivan, L. S., Ray, J. W., Daiger, S. P., Birch, D. G., and Hughbanks-Wheaton, D. K. (2011), "Phenotypic Characterization of 3 Families With Autosomal Dominant Retinitis Pigmentosa Due to Mutations in KLHL7," *Archives of Ophthalmology*, 129, 1475–1482. [1526]

Yao, W., Nandy, D., Lindsay, B. G., and Chiaromonte, F. (2019), "Covariate Information Matrix for Sufficient Dimension Reduction," *Journal of the American Statistical Association*, 114, 1752–1764. [1517,1518,1519,1527]

Zhang, C.-H. (2010), "Nearly Unbiased Variable Selection Under Minimax Concave Penalty," *The Annals of Statistics*, 38, 894–942. [1516]

Zhong, W., and Zhu, L. (2015), "An Iterative Approach to Distance Correlation-Based Sure Independence Screening," *Journal of Statistical Computation and Simulation*, 85, 2331–2345. [1528]

Zhu, L.-P., Li, L., Li, R., and Zhu, L.-X. (2011), "Model-Free Feature Screening for Ultrahigh-Dimensional Data," *Journal of the American Statistical Association*, 106, 1464–1475. [1517,1519,1520,1521,1522,1523,1528]

Zou, H., and Hastie, T. (2005), "Regularization and Variable Selection via the Elastic Net," *Journal of the Royal Statistical Society*, Series B, 67, 301–320. [1516]