Sure Independence Screening for Ultra-High Dimensional Feature Space

Mänd Lv (2008), JRSS-B

https://doi.org/10.1111/j.1467-9868.2008.00674.x

Stat 227 Project - Zach Horton

6 May, 2021

Ultra-High Dimensional Data

Assume the linear regression framework:

$$\underset{n\times 1}{\boldsymbol{y}} = \underset{n\times p}{X}\underset{p\times 1}{\boldsymbol{\beta}} + \underset{n\times 1}{\boldsymbol{\epsilon}}$$

In ultra-high dimensional settings where $p >> n$ several issues arise:

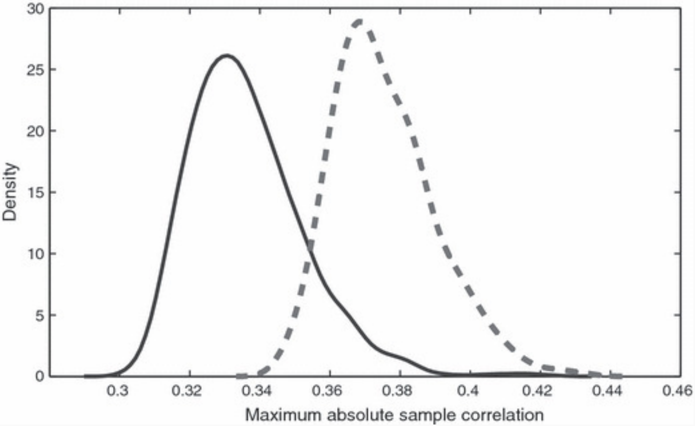
- The design matrix X is rectangular $\implies X^T X$ is huge and singular
- The covariance Σ may become ill-conditioned as n grows, making variable selection difficult
- The minimum coefficient $|\beta_j|$ may decay with n to the level of noise
- Decorrelation by $L^{-1}\boldsymbol{y}$ may produce heavy tails
- Shrinkage, best subset selection, and other standard dimensionality reduction techniques often struggle either in stability or in computation time

Also, spurious correlations...

Ultra-High Dimensional Data

Spurious correlations:

- $n = 60$ observations
- $p = 1000$ variables (solid)
- $p = 5000$ variables (dashed)



Variable Screening

Suppose there is a “true” model $\boldsymbol{y} = X_T \boldsymbol{\beta}_T + \epsilon$ with each $\beta_{T_j} \neq 0$:

- Assume the true factors are contained $X_T \subset X_D$ in the data
- Let $p_T = |\boldsymbol{\beta}_T|$ be the true model size

Variable Screening

Suppose there is a “true” model $\boldsymbol{y} = X_T \boldsymbol{\beta}_T + \epsilon$ with each $\beta_{T_j} \neq 0$:

- Assume the true factors are contained $X_T \subset X_D$ in the data
- Let $p_T = |\boldsymbol{\beta}_T|$ be the true model size

A desirable screening technique which selects a set of ”screened” factors X_S ideally satisfies:

- **Independence screening**: selecting each variable without consideration of others
- **Sure screening property**: $\lim_{n \to \infty} P(X_S \subset X_D) = 1$

Sure Independence Screening

The Sure Independence Screening (SIS) algorithm is one of the main aspects of Fan and Lv (2008):

1. Compute $\boldsymbol{\omega} = X_D^T \boldsymbol{y}$ the ”component-wise regression” coefficients
2. Sort elements of $\boldsymbol{\omega}$ by absolute magnitude
3. Retain the largest $d < n$ variables where $d = \lceil \gamma n \rceil$, $\gamma \in (0, 1)$

This algorithm is simple, fast, and satisfies the sure screening property![1]

[1] SIS is intended to be used as pre-processing before using more common analysis tools. Thus d is chosen such that it produces reasonable amounts of data for subsequent methods. Reasonable choices include $n - 1$ and $\frac{n}{\log(n)}$.

SIS Assumptions

Main assumptions proving that SIS has the sure screening property:

1. The true model specification is correct (linear, contained in the data)
2. Observations $\boldsymbol{x}_D$ arise independently from a spherically symmetrical distribution[2]
3. The eigenvalues of Σ have certain lower and upper limiting bounds (concentration property)

[2] The p-variate normal is a common case. Also note the authors made no attempt to relax this or determine minimally sufficient conditions.

SIS Assumptions

Main assumptions proving that SIS has the sure screening property:

1. The true model specification is correct (linear, contained in the data)
2. Observations $\boldsymbol{x}_D$ arise independently from a spherically symmetrical distribution[2]
3. The eigenvalues of Σ have certain lower and upper limiting bounds (concentration property)

Popular linear modeling techniques may violate condition 2.

- Binary/categorical variables
- Interactions terms
- Polynomial basis expansions

This project explores empirical/simulated performance of SIS under these conditions with respect to the sure screening property.

[2] The p-variate normal is a common case. Also note the authors made no attempt to relax this or determine minimally sufficient conditions.

Simulation Study Setup

Basic simulation outline:

1. Generate n observations $\boldsymbol{x}_i$ from a $N_{p^*}(\mathbf{0}, \Sigma)$. We set $\Sigma = \rho \mathbf{1}\mathbf{1}^T + (1-\rho) I_{p^*}$
2. Expand or transform according to the setting. Note p^* is chosen to form p variables after expansion
3. Collect the results to form the $n \times p$ data matrix X_D
4. Randomly select 4 column indices T_1, T_2, T_3, T_4
5. Construct observations $\boldsymbol{y} = 5\boldsymbol{x}_{T_1} + 5\boldsymbol{x}_{T_2} + 5\boldsymbol{x}_{T_3} + 5\boldsymbol{x}_{T_4} + \boldsymbol{\epsilon}$, with $\boldsymbol{\epsilon} \sim N_n(\mathbf{0}, 0.01 I_n)$
6. Apply SIS to to X_D and $\boldsymbol{y}$ with $d = 2\frac{n}{\log(n)}$
7. Determine if $X_T \subset X_{SIS}$
8. Repeat steps 1-7 for each scenario 1000 times

Scenarios consist of all combinations of $p = 500, 1000, 2000$, $n = 20, 30, 50, 100$, $\rho = 0, 0.1, 0.5, 0.9$. We do this for each setting: standard, binary variables, interactions, and quadratic expansion.

Simulation Results - Standard Case

Generate $p^* = p$ dimensional observations. No transformation/expansion.

p	n	$\rho = 0$	$\rho = 0.1$	$\rho = 0.5$	$\rho = 0.9$
500	20	0.023	0.022	0.025	0.027
	30	0.218	0.221	0.217	0.210
	50	0.809	0.816	0.793	0.775
	100	0.998	1.000	0.997	0.992
1000	20	0.008	0.007	0.010	0.006
	30	0.092	0.110	0.106	0.086
	50	0.681	0.651	0.650	0.658
	100	0.999	0.993	0.998	0.993
2000	20	0.003	0.001	0.004	0.004
	30	0.046	0.032	0.038	0.034
	50	0.499	0.486	0.499	0.527
	100	0.995	0.991	0.988	0.987

Simulation Results - Binary Case

Generate $p^* = p$ dimensional observations. Randomly select half and convert to $(-1, 1)$ binary.

p	n	$\rho = 0$	$\rho = 0.1$	$\rho = 0.5$	$\rho = 0.9$
500	20	0.039	0.039	0.006	0.003
	30	0.307	0.284	0.094	0.024
	50	0.862	0.855	0.553	0.060
	100	0.997	1.000	.986	0.074
1000	20	0.018	0.008	0.001	0.004
	30	0.125	0.138	0.028	0.007
	50	0.760	0.754	0.329	0.045
	100	1.000	0.999	0.952	0.055
2000	20	0.002	0.003	0.000	0.000
	30	0.047	0.063	0.002	0.007
	50	0.604	0.585	0.174	0.041
	100	0.995	0.993	0.890	0.057

Simulation Results - Interaction Case

Generate p^* dimensional observations such that the total pairwise interaction terms plus the original variables is as near to p as possible.

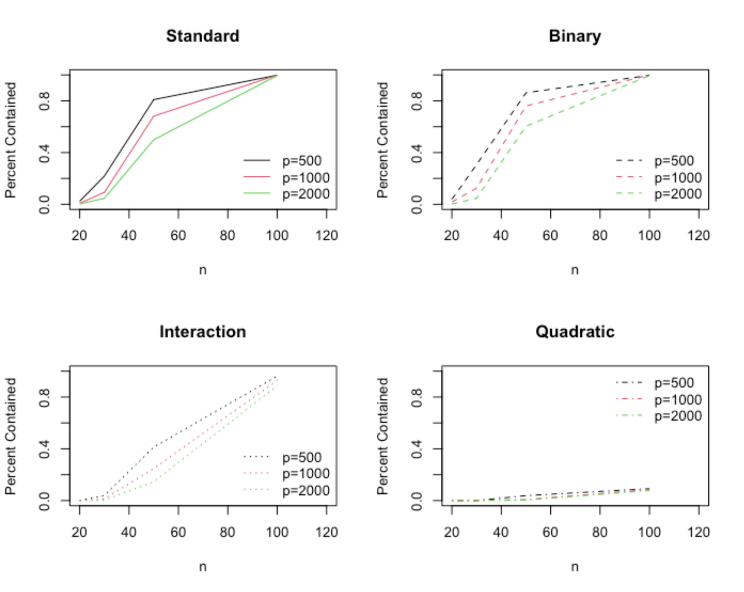
p	n	$\rho = 0$	$\rho = 0.1$	$\rho = 0.5$	$\rho = 0.9$
500	20	0.002	0.001	0.001	0.000
	30	0.039	0.029	0.032	0.020
	50	0.413	0.413	0.314	0.241
	100	0.962	0.965	0.915	0.569
1000	20	0.000	0.000	0.000	0.001
	30	0.013	0.011	0.011	0.008
	50	0.248	0.284	0.219	0.161
	100	0.928	0.937	0.872	0.646
2000	20	0.000	0.000	0.000	0.000
	30	0.003	0.002	0.002	0.005
	50	0.146	0.140	0.117	0.114
	100	0.887	0.879	0.812	0.672

Simulation Results - Polynomial Case

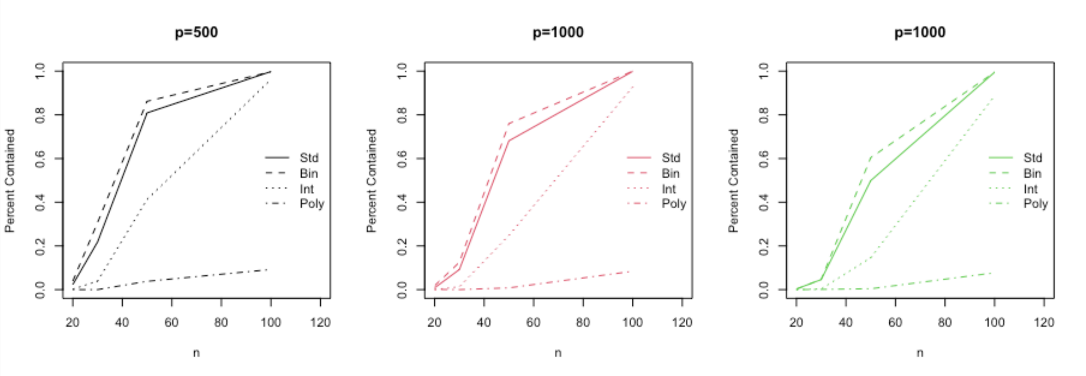
Generate $p^* = p/2$ dimensional observations. Expand by adding the squared version of each variable.

p	n	$\rho = 0$	$\rho = 0.1$	$\rho = 0.5$	$\rho = 0.9$
500	20	0.000	0.001	0.000	0.000
	30	0.000	0.000	0.001	0.000
	50	0.038	0.023	0.008	0.005
	100	0.092	0.102	0.079	0.034
1000	20	0.001	0.000	0.000	0.000
	30	0.000	0.000	0.000	0.000
	50	0.008	0.003	0.002	0.001
	100	0.084	0.066	0.047	0.024
2000	20	0.000	0.000	0.000	0.000
	30	0.001	0.000	0.000	0.000
	50	0.004	0.003	0.001	0.001
	100	0.077	0.054	0.025	0.009

Simulation Results



Simulation Results



Conclusions

Main findings:

- Binary predictors do well with SIS
- Interactions are moderately successful
- Polynomial expansions fail in comparison

Side notes:

- Binary predictors struggle with large ρ (may be data artifact)
- Interactions seem more influenced by ρ than others
- Larger p slightly decreases convergence rate

Future ideas:

- Test combinations of the settings
- Test using the Iterative SIS algorithm
- Test different d values and the associated costs/benefits