**Statistica Sinica Preprint No: SS-2021-0187**

<table>
<thead>
<tr>
<th><strong>Title</strong></th>
<th>Ensemble Subset Regression (ENSURE): Efficient High-dimensional Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Manuscript ID</strong></td>
<td>SS-2021-0187</td>
</tr>
<tr>
<td><strong>URL</strong></td>
<td><a href="http://www.stat.sinica.edu.tw/statistica/">http://www.stat.sinica.edu.tw/statistica/</a></td>
</tr>
<tr>
<td><strong>DOI</strong></td>
<td>10.5705/ss.202021.0187</td>
</tr>
</tbody>
</table>
| **Complete List of Authors** | Rong Zhu  
                             Hua Liang and  
                             David Ruppert |
| **Corresponding Author** | Hua Liang                  |
| **E-mail**      | hliang@gwu.edu                                                          |

Notice: Accepted version subject to English editing.
Ensemble Subset Regression (ENSURE): Efficient High-dimensional Prediction

Rong Zhu, Hua Liang and David Ruppert

Fudan University, George Washington University, and Cornell University

Abstract: In high-dimensional prediction problems, we propose to subsample the predictors prior to analysis. Specifically, we draw features by random sampling, and then fit a model and make predictions based on the sampled feature subset. This greatly reduces the dimension, storage, and computational bottlenecks. We explore this "subset regression" strategy under the linear regression framework. More interestingly, an ensemble method by combining multiple subset regressions, called Ensemble Subset Regression (ENSURE), is proposed here to reduce the uncertainty due to feature sampling. Theoretically, we provide an upper bound on the excess risk of the predictions computed in the subset regression and provide theoretical support that the ensemble can improve the performance of the subset regression. Detailed empirical studies also demonstrate that ENSURE performs well, better than methods using all features.

Key words and phrases: Feature subset; High-dimensional; Non-sparsity; Random sampling; Ridge regression; Uniform sampling.
1. Introduction

High-dimensional problems have arisen from diverse scientific areas and various statistical methods have been developed to deal with such problems. In this article, we study a new method for predicting an outcome variable $Y$ from feature variables $X_1, X_2, \ldots, X_p$ measured on each of $n$ individuals. We are interested in the high-dimensional scenario where the dimension $p$ is much larger than the sample size $n$.

However, high-dimensional prediction runs into the curse of dimensionality. Classic strategies make use of techniques to identify important variables or components in order to improve the prediction efficiency. These techniques include best subset selection methods like the nonnegative garrote (Breiman and Spector 1992, Breiman 1995), penalization based methods such as ridge regression (Hoerl 1962, Hoerl and Kennard 1970), least absolute shrinkage and selection operator (LASSO, Tibshirani 1996), and its variants (Fan and Li 2001, Yuan and Lin 2006, Efron et al. 2004, Zou and Hastie 2005), Dantzig selector (Candes and Tao 2007), and principal components (Bair et al. 2006). It is well known that prediction is one of the fundamental concepts and aims in machine learning, statistics, and other disciplines. Besides prediction, identification of important variables and model selection is often desirable. See Hastie et al. (2009) for a review. Here, however, we investigate prediction rather than the variable selection, since prediction has its independent interest. Recently,
prediction performance of high-dimensional modeling has been given a substantial amount of attention. Cook et al. (2013) discussed asymptotic characteristics of prediction in high-dimensional linear regression. Dalalyan et al. (2017) investigated the relationship between the prediction performance and the correlations of the covariates for LASSO.

In this paper, we propose a new approach to high-dimensional prediction based on random sampling of the predictors prior to analysis and performing the $l_2$-regularized linear regression using this subset. This strategy, which we call “subset regression,” uses the survey sampling principle that one can obtain an accurate estimate from a subsample (a proxy of the finite-population-based estimate) taken by random sampling from a finite population. From the subset regression, we get the approximate prediction that can substitute for the full high-dimensional prediction. Compared with methods using full features, the subset regression can greatly reduce the computational cost, and also gets close to the prediction using all features. Theoretically, we provide an upper bound on the excess risk of prediction from subset regression. This excess risk bound establishes convergence rate for the prediction from the subset regression with respect to the true prediction.

More interestingly, we apply the ensemble method to subset regression to reduce the uncertainty due to feature sampling. This is accomplished by generating multiple subsets, from each of which a prediction is obtained, and then using the
ensemble method to average these results. Our theoretical results elucidate the effect on performance of applying the ensemble method, and show that the excess risk can be reduced by the ensemble method. Empirical studies strikingly indicate that coupling the subset regression with its ensemble method typically yields much better prediction than the methods based on full features.

Our method is related to Bagging (or its modification Random Forests) (Breiman, 1996, 1998; Ho, 1998; Breiman, 2001; Brylla et al., 2003), which fits the same regression model many times to bootstrap-sampled versions of the training data, and averages the results. However, our method gets multiple predictions via sampling features, rather than bootstrapping the training data as Bagging does. This paper demonstrates the potential benefit of the feature sampling ensemble for high-dimensional regression problem. Our contribution is firstly to show that predictions using a single subset regression perform well and then that, under ensemble learning, subset regression can further improve prediction efficiency. Empirically, our Ensemble Subset Regression (ENSURE) often beats previously proposed high-dimensional methods if our aim is accurate prediction. Our studies indicate that, if we combine multiple predictions, then it may be not necessary to use all the features in a high-dimensional dataset.

There is substantial parallel work on efficiently analyzing high-dimensional problems by using random projections, which are used to well approximate a regression
function in a linear space of high dimension using projections onto a random subspace of lower dimension. Maillard and Munos (2009) proposed compressed least-squares regression, which is further studied by Fard et al. (2012). Guhaniyogi and Dunson (2015) proposed the Bayesian version of compressed least-squares regression. In addition, this high-dimension projection is also used in other methods, such as SVM (Krishnan et al., 2007), and discriminant analysis (Cannings and Samworth, 2017). However, this article offers an alternative way by feature sampling, and, as far as we know, is the first work on random sampling into the high-dimensional prediction in linear models.

The remainder of the paper is organized as follows. Section 2 introduces the subset regression by feature sampling. We provide a bound on excess risk for subset regression in Section 3. An ensemble method for the subset regression is proposed in Section 4. We report simulation results in Section 5, where the performance of subset regression is compared with that of methods using full features. Further discussion and concluding remarks are given in Section 6. Proofs of the theoretical results are in the Appendix.

Notations. Throughout this paper, for a matrix $X \in \mathbb{R}^{n \times p}$, $x^{(k)}$ represents the $k$th column of $X$, $x_i$ represents the $i$th row of $X$, and $x_{ik}$ is the $(i,k)$th element of $X$. For a vector $\beta \in \mathbb{R}^p$, we define $\beta_k$ as the $k$th element of $\beta$. 
2. Subset Regression by Feature Sampling

In this section, we first introduce feature sampling, and study the convergence properties of the feature sampling for high-dimensional prediction. Then, we propose a subset regression algorithm for the high-dimensional prediction.

2.1 Problem Setting

In such a setting, it is of great interest to study the high-dimensional prediction problem for the linear regression model:

\[ y_i = \mu_i + \epsilon_i, \]  
\[ \mu_i = x_i^\top \beta = \sum_{k=1}^{p} x_{ik} \beta_k, \]  
\[ \{ \epsilon_i \}_{i=1}^{n} \] are the model errors with zero mean and \( \sigma^2 \) variance, and \( \beta \in \mathbb{R}^p \) is a parameter vector. We allow \( p \gg n \) and \( \beta \) to be non-sparse. Suppose that there is a finite set of training examples \( \{(x_i, y_i)\}_{i=1}^{n} \) from model (2.1). We assume that, without a loss of generality, the inputs \( \{x_i\}_{i=1}^{n} \) and the output \( \{y_i\}_{i=1}^{n} \) have been centered. The matrix form of the model (2.1) is

\[ y = \mu + \epsilon = X\beta + \epsilon, \]
where \( \mu = (\mu_1, \cdots, \mu_n)^\top \), \( \epsilon = (\epsilon_1, \cdots, \epsilon_n)^\top \) and \( X = (x_1^\top, \cdots, x_n^\top)^\top \). In this article, we focus on a canonical instance of a high-dimensional prediction problem, namely predicting \( \mu_x = x^\top \beta \) given an input \( x \), rather than estimating \( \beta \).

### 2.2 Feature Sampling

The idea of feature sampling is to approximate the summation term \( \sum_{k=1}^{p} x_{ik} \beta_k \) in Eqn. (2.2) with a summation based on a feature subset obtained by random sampling. Generate a feature subset \( S \subset U = \{1, \ldots, p\} \) of size \( p_s \) (\( p_s = |S| \)) by uniform sampling without replacement from full features \( \{1, \ldots, p\} \), i.e., we draw \( X_s = \{x^{(k)}, k \in S\} \) from \( X \). Note that we apply sampling without replacement rather than with replacement, since sampling with replacement may be less efficient than sampling without replacement \( \text{\cite{Sarndal2003}} \) and may induce additional collinearity. However, our analysis below indicates that sampling with replacement is applicable in principle for the subset regression, and its theoretical analysis is easier than that of sampling without replacement.

The feature sampling can be considered as a standard survey sampling procedure that draws a “sample set” (subset) \( S \) of size \( p_s \) from the population \( \{x_i \beta_1, \ldots, x_i \beta_p\} \) by uniform sampling without replacement. The sampling ratio be \( f_s = p_s / p \). We estimate the population total in (2.2) from the subset \( S \)

\[
\mu_{is} = \sum_{k \in S} f_s^{-1} x_{ik} \beta_k = f_s^{-1} x_{is}^\top \beta_s,
\] (2.4)
where $x_{is}$ and $\beta_s$ are subsets of $x_i$ and $\beta$, respectively, corresponding to the subset $S$. Note that $\mu_{is}$ is inaccessible in practice because $\beta$ is unknown. However there is an intuitive that the prediction based on feature sampling potentially works well when $\mu_{is}$ can approximate $\mu_i$. We will further explore this approximation theoretically. It is easy to verify that $E(\mu_{is}) = \mu_i$, and

$$\text{Var}(\mu_{is}) = \frac{1 - f_s}{p_s} \sum_{k=1}^{p} x_{ik}^2 \beta_k^2 < \frac{1}{f_s} \sum_{k=1}^{p} x_{ik}^2 \beta_k^2. \quad (2.5)$$

From Markov’s inequality, we directly get the following theorem.

**Theorem 1.** We have that

$$|\mu_{is} - \mu_i| \leq \rho^{-1} p_s^{-1/2} p^{1/2} \left( \sum_{k=1}^{p} x_{ik}^2 \beta_k^2 \right)^{1/2} \quad (2.6)$$

with probability at least $1 - \rho$.

Theorem 1 indicates that the excess error of $\mu_{is}$ can be bounded by $p_s^{-1/2} p^{1/2} \left( \sum_{k=1}^{p} x_{ik}^2 \beta_k^2 \right)^{1/2}$. Therefore, we define the condition that $\mu_i$ is $c$-compatible if there exists a constant $c \geq 0$ such that

$$\sum_{k=1}^{p} x_{ik}^2 \beta_k^2 \leq cp_{is} \delta^{-1} |\mu_i|^2, \text{ for some } 0 \leq \delta < 1. \quad (2.7)$$

From (2.6), as $p_s \to \infty$,

$$\frac{|\mu_{is} - \mu_i|}{|\mu_i|} = o_p(1), \quad (2.8)$$

where we assume $|\mu_i| \neq 0$. Eqn. (2.8) shows that $\mu_{is}$ can approximate $\mu_i$ well. Notice that $|\mu_i|^2 \leq p \sum_{k=1}^{p} x_{ik}^2 \beta_k^2$ by the Cauchy-Schwartz inequality, and the equality holds
if and only if $x_{ik} \beta_k$ are completely homogeneous. Roughly speaking, Condition (2.7) excludes some extreme sparsity cases, but the term $p_s^\delta$ in Condition (2.7) allows a certain degree of heterogeneity of the contribution of each feature to the prediction. It is worth noting that our excess risk bound provided in Section 3 does not rely on Theorem 1. Instead Theorem 1 provides a valuable hint that the subset prediction may be reasonable.

**Remark.** We perform *uniform sampling* for its computational simplicity and statistical efficiency, the latter of which is numerically verified in Section 5. However, *uniform* sampling may lose statistical efficiency relative to data-driven importance sampling. It would be very interesting to develop a more efficient data-based importance sampling method, since it does makes sense to put more weight on the features that carry more information or is more important.

### 2.3 Prediction on Feature Subset

From Theorem 1, we know that it may make sense to replace $\mu = \sum_{k \in U} x^{(k)} \beta_k$ with the approximation $\tilde{\mu} = f_s^{-1} \sum_{k \in S} x^{(k)} \beta_k$ to construct the least-squares estimate based on the feature subset. Note: the goal of this paper is prediction, so we ignore the constant $f_s$, which can be absorbed into $\beta_k$, and simply rewrite the approximation as $\tilde{\mu} = \sum_{k \in S} x^{(k)} \beta_k$ without loss of generality. Meanwhile, note that the subset size $p_s$ may be not much smaller, but typically is larger than the sample size $n$, so we
add a $l_2$ regularized penalty to subset regression, that is, we minimize

$$\text{RSS}_S(\beta_s) = \|y - \sum_{k \in S} x^{(k)} \beta_k\|^2 + \lambda \sum_{k \in S} \beta_k^2,$$

where $\lambda > 0$ is a penalty parameter. This is a statement about the feature subset that we draw by sampling without replacement.

By minimizing the function $\text{RSS}_S(\beta_s)$ based on the feature subset, we get an estimator $\hat{\beta}_{s,RR}$:

$$\hat{\beta}_{s,RR} = (X_s^T X_s + \lambda I_s)^{-1} X_s^T y.$$

where we use generalized cross-validation (GCV) to choose the regularized parameter $\lambda$. From (2.4) and (2.10), the prediction of $\mu_i$ is

$$\hat{\mu}_i = \sum_{k \in S} x_{ik} \hat{\beta}_{s,k},$$

where $\hat{\beta}_{s,k}$ is the $k$th element of $\hat{\beta}_{s,RR}$. Thus, the approximate fitted values of $\mu = (\mu_1, \cdots, \mu_n)^T$ is

$$\hat{\mu}_s = \sum_{k \in S} x^{(k)} \hat{\beta}_{s,k} = X_s \hat{\beta}_{s,RR}$$

$$= X_s \left((X_s^T X_s + \lambda I_s)^{-1} X_s^T y\right).$$

In brief, our proposed subset regression is presented in Algorithm 1 below.

**Remark.** We use $l_2$-regularization (“ridge”) in our subset regression because of its simplification and nice performance. We shall show the nice performance of our
Algorithm 1 Subset Regression Algorithm

- **Random Sampling Features**: Draw a feature subset by uniform sampling from full features.

- **Estimation**: Solve the ridge regression problem using the feature subset, i.e., minimize the function $\text{RSS}_S(\beta_s)$ in Eqn.(2.9) to get an estimate $\hat{\beta}_{s,RR}$.

- **Prediction**: Calculate the prediction values $\hat{\mu}$ using Eqn.(2.11) from the feature subset.

subset regression with $l_2$-regularization in Sections 3 & 5 from the theoretical and empirical perspectives, respectively. The regularization can reduce mean squared error by potentially allowing a slight increase in bias but dramatically reducing the variance. Theoretically, Shao and Deng (2012) investigated the consistency of the ridge estimator in high-dimensional setting. The regression with $l_2$-regularization also has been proved to be effective in many applications. This improvement is remarkable in predictive performance (Frank and Friedman 1993, Malo et al. 2008). Tibshirani (1996) and Fu (1998) compared predictions from LASSO and ridge regression and found that the latter is competitive even in some sparse settings.
3. An Upper Bound on Excess Risk

In this section, we provide an upper bound on the excess risk of the subset regression,

\[ R(\hat{\mu}_s) = E\|\hat{\mu}_s - \mu\|^2. \]

We re-express \( \hat{\mu}_s \) by applying the Sherman-Morrison-Woodbury update. Recall this classical method for a low-rank update of an inverse of a matrix: for \( V \in \mathbb{R}^{n \times m} \) and \( D \in \mathbb{R}^{m \times m} \),

\[ (V^TV + D)^{-1} = D^{-1} - D^{-1}V^T(VD^{-1}V^T + I)^{-1}VD^{-1}. \]

We apply this equality into \((X_s^TX_s + \lambda I_n)^{-1}\) and then get the equation:

\[
(X_s^TX_s + \lambda I_n)^{-1} = \lambda^{-1}I_n - \lambda^{-1}X_s^T(X_s^TX_s + \lambda I_n)^{-1}X_s. \tag{3.12}
\]

By inserting (3.12) into (2.11), we have

\[
\hat{\mu}_s - \mu = (X_sX_s^T + \lambda I_n)^{-1}X_sX_s^T(X\beta + \epsilon) - X\beta \\
= -\lambda(X_sX_s^T + \lambda I_n)^{-1}X\beta + (X_sX_s^T + \lambda I_n)^{-1}X_sX_s^T\epsilon \\
= :d_B + d_V. \tag{3.13}
\]

The term \( d_B = -\lambda(X_sX_s^T + \lambda I_n)^{-1}X\beta \) is the bias of the subset regression prediction\

\( d_V = (X_sX_s^T + \lambda I_n)^{-1}X_sX_s^T\epsilon \) is the noise that determines the variance.

In the following, we investigate the performance of the subset regression as \( \lambda \to 0; \)
i.e.,

\[
d_B \to -[I_n - X_s(X_s^TX_s)^+X_s^T]X\beta, \text{ and } \\
d_V \to X_s(X_s^TX_s)^+X_s^T\epsilon.
\]
Note that the bias term of the ridge regression based on full features goes to 0 as $\lambda \to 0$, but the bias $d_B$ of our subset regression does not. This indicates that compared to the ridge method from full features, the error caused by the bias in the subset regression would be larger. On the other hand, the error caused by the variance in the subset regression may decrease since the dimension decreases from $p$ to $p_s$.

Based on the decomposition of Eqn. (3.13), we are now ready to present our main theoretical result, that is, an upper bound on the excess risk of $\hat{\mu}_s$.

**Theorem 2.** Assume that $X$ is full row rank and denote $d_1 \geq \cdots \geq d_n > 0$ as the ordered non-zero eigenvalues of $p^{-1}XX^\top$. Define $a = \max\{\|x^{(k)}\|^2\}_{k=1}^p$. Let $1 > \rho, \kappa > 0$. A feature subset of size $p_s$ is randomly drawn from all $p$ features by uniform sampling without replacement. If

$$p_s > \left(\ln \frac{n}{\rho}\right) \frac{\kappa/3 + ad^{-1}}{\kappa^2/2},$$

then

$$\mathcal{R}(\hat{\mu}_s) \leq \frac{1}{(1 - \kappa)^2} \left[ \sum_{i=1}^{n} \frac{\lambda^2 \mu_i^2}{(pd_i + \lambda)^2} + \sigma^2 \sum_{i=1}^{n} \frac{pd_i}{pd_i + \lambda} \right]$$

with probability at least $1 - \rho$.

Now we take a look at the bound on $\mathcal{R}(\hat{\mu}_s)$ in Theorem 2. The bound has three terms: the risk from model approximation, represented by $\sum_{i=1}^{n} \frac{\lambda^2 \mu_i^2}{(pd_i + \lambda)^2}$, the risk from
the noise, represented by \( \sigma^2 \sum_i n \frac{p d_i}{p d_i + \lambda} \), and the risk from the subsampling, represented by \( (1 - \kappa)^{-2} \). The first two terms are the risk from the ridge regression based on full features. When \( \lambda \) goes to 0, the first term becomes small, while the second term becomes large. The third term \( (1 - \kappa)^{-2} \) controls the extra risk brought from the subsampling step. Therefore, Theorem 2 provides a theoretical foundation for the subset regression in order to show that the extra risk can be small when the feature subset size \( p_s \) is large enough. This property indicates that the computationally cheap algorithm is also statistically efficient for high-dimensional prediction. Next we take a look at the requirement on \( p_s \) in Eqn. (3.14). Assuming each element in \( X \) is bounded, we have \( a = \max \{ \| x^{(k)} \|^2 \}_{k=1}^p = O(n) \). Assuming that each \( x_i \) is independent and identically distributed, \( p^{-1} X X^\top \) is close to a diagonal matrix which has the same diagonal element. It implies that \( d_i \) and \( d_i^{-1} \) are constants of order \( O(1) \). Thus, in Eqn. (3.14), given constants \( \kappa \) and \( \rho \), \( (\ln \frac{n}{\rho})^{\kappa/3 + \alpha d_i^{-1}} = O(\log n) \), implying that the requirement on \( p_s \) can be still much smaller than \( p \) given the fact that \( n \ll p \) in our high-dimensional setting. As a result, the feature subset size \( p_s \) that satisfies Eqn. (3.14) can be dramatically smaller than \( p \) of the original feature.

In terms of choice of \( p_s \), we provide a cross-validation strategy, whose numerical performance is investigated in the empirical studies below. Specifically speaking, equation (3.14) provides guidance about the relationship between the feature subset size and the approximation accuracy from a theoretical point of view; i.e., \( p_s =\)
$O(n \log n)$. Empirically from simulated and real data sets in Section 5 the risk based on the subset regression procedure can approach that based on the full model as $p_s$ increases. Therefore, we suggest choosing $p_s$ using $p_s = cn$ where $c$ is some integer which can be simply chosen by validation in practice.

**Remark.** Comparing Theorems 2 with 1, Theorem 2 does not require the result in Theorem 1 that is, we do not require the term $\sum_{k=1}^p \|x^{(k)}\|^2 \beta_k^2$ be as small as in Theorem 1. The condition in Theorem 2 is that the subset size $p_s$ need to be large enough compared to the sample size $n$.

**Remark.** Although we choose simple uniform sampling for subset regression, unequal sampling dependent on $X$ is also applicable for the proposed subset regression and the excess risk analysis. Specifically, assume that we sample the subset $S$ by unequal sampling without replacement from full features proportional to the sampling probabilities $\{\pi_1, \cdots, \pi_p\}$ such that $\sum_{k=1}^p \pi_k = 1$. The objective function Eqn.(2.9) is replaced by

$$\text{RSS}_S(\beta_s) = \|y - \sum_{k \in S} \frac{1}{\pi_k} x^{(k)} \beta_k\|^2 + \lambda \sum_{k \in S} \frac{1}{\pi_k} \beta_k^2.$$ 

Following the process of the subset regression we get the approximate prediction of $\mu$,

$$\hat{\mu}_s = X_s (X_s^T X_s + \lambda \Phi_s)^{-1} X_s^T y,$$

where $\Phi_s$ is the submatrix of $\Phi = \text{diag}\{\pi_1, \cdots, \pi_p\}$ corresponding to the sub-
set $S$. Moreover, Theorem 2 still holds when $p_s > (\ln \frac{n}{\rho})^{\kappa/3 + \frac{Ad}{n}}$, where $A = \max\{(p\pi_k)^{-1}\|x^{(k)}\|_2^2\}_{k=1}^p$. We supply the proofs in Appendix A.1.2.

4. Subset-ensemble Prediction

In this section, we show that a mixture of subset regression predictions, which we call subset-ensemble prediction, yields a more accurate prediction than a single subset regression. This result is of great practical significance given the prevalence of distributed computing frameworks in handling large-scale learning problems. The ensemble algorithm naturally fits the distributed computing environment where their computational cost is roughly the same as that of the standard subset regression. The algorithm is very simple and easily implemented. Assume that there are $T$ predictions from subset regression, treat each prediction from as an expert, and combine $T$ experts to get an improved prediction. We present the ensemble process in Algorithm 2 below.

From Algorithm 2 we get the risk of $\hat{\mu}_{\text{ensure}}$.

$$R(\hat{\mu}_{\text{ensure}}) = E\|\hat{\mu}_{\text{ensure}} - \mu\|_2^2 = E\|\frac{1}{T} \sum_{t=1}^T (\hat{\mu}_{s,t} - \mu)\|_2^2$$

We provide a comparison between the risks $R(\hat{\mu}_{\text{ensure}})$ and $R(\hat{\mu}_s)$ in the following theorem.
Algorithm 2 Ensemble Subset Regression Algorithm

- **Repeat**: Repeat the following steps $T$ times:

  (a) **Draw a feature subset**: A feature subset of size $p_s$ is sampled from the full features;

  (b) **Make a single prediction**: A prediction from subset regression has output $\hat{\mu}_{s,t}$;

- **Combine**: Majority vote by using the equation below.

$$\hat{\mu}_{\text{ensure}} = T^{-1} \sum_{t=1}^{T} \hat{\mu}_{s,t},$$

(4.15)

**Theorem 3.** Denote $\sigma_B^2 = E\|d_B\|^2 - \|E(d_B)\|^2$ and $\sigma_V^2 = E(\|d_V\|^2|y) - \|E(d_V|y)\|^2$.

$$\mathcal{R}(\hat{\mu}_s) - \mathcal{R}(\hat{\mu}_{s,\text{ensure}}) = (1 - \frac{1}{T})\{\sigma_B^2 + E_m(\sigma_V^2)\},$$

(4.16)

where “$E_m()$” refers to the expectation over the model (2.1).

Theorem 3 tells us an interesting observation about the subset-ensemble method.

The risk performance depends on three factors: $\sigma_B^2$ from $d_B$, $\sigma_V^2$ from $d_V$, and the ensemble number $T$. $\sigma_B^2$ denotes the variability of $\|d_B\|$ from the feature sampling; and $\sigma_V^2$ denotes, fixing the responses $y$, the variability of $\|d_V\|$ from the feature sampling.

Due to introducing ensemble, the excess risk decreases by $(1 - \frac{1}{T})\{\sigma_B^2 + E_m(\sigma_V^2)\}$. In other words, the ensemble algorithm can improve the performance by reducing the risk from these variabilities of $\|d_B\|$ and $\|d_V\|$. From the computational viewpoint,
although the subset-ensemble prediction requires $T$ times more time than the single subset regression, we note that the ensemble version can be easily parallelized, as all $T$ experts can be computed simultaneously. Thus, for a cluster of $T$ machines, the running time complexity of subset-ensemble prediction is closely equal to that of the single subset regression.

**Choice of $T$.** Theorem 3 shows that the benefit due to the ensemble would narrow as $T$ increases. Empirical studies also verify this observation. In practice, it is enough to set $T = 10$ such that Algorithm 2 has a sufficiently nice prediction.

5. Numerical Experiments

In this section, we conduct detailed experiments for assessing the performance of the subset regression. We compare ensemble subset regression (ENSURE) with several representative high-dimensional methods: ridge regression (ridge), LASSO, sure independence screening (SIS), and random forests (RF). We implement LASSO, SIS, and RF by the R packages *glmnet* [Friedman et al., 2010], *SIS* [Fan and Lv, 2008], and *randomForest* [Breiman, 2001], respectively. Default settings are used in the implementation of these methods. In the implementation of SIS, we use the SCAD penalty (default) and tune the regularization parameter by BIC. The penalizing parameters for subset regression and ridge regression were selected by GCV.
In Section 5.1 we report the results of extensive simulation experiments. In Section 5.2 we apply ENSURE to two real datasets. In both the simulations and in the real data examples, our subset-ensemble method outperforms methods based on full features.

5.1 Simulation Studies

We generated data from the linear model $y_i = \mu_i + \epsilon_i$ with $\mu_i = 1 + x_i^\top \beta$ for $i = 1, \ldots, n$, where $\epsilon_i \sim N(0, \sigma^2)$. For evaluation purposes, we consider two design matrix settings. (1) autoregressive correlation $x_i \sim N(0, \Sigma)$, where $\Sigma_{ij} = \rho^{|i-j|}$ (referred as to AR data set). (2) we also investigate an input matrix from a heavier tail, i.e., $x_i$ is independently from $t_2$ distribution (referred as to $T_2$ data set). For specific, $x_i = \frac{z_i}{\sqrt{u}}$ where $z_i \sim N(0, \Sigma)$ independent of $2u \sim \chi^2_2$. To control the signal-to-noise ratio, we define $\sigma^2 = \frac{1}{n-1} \sum_{i=1}^{n} (\mu_i - \bar{\mu})^2 / r$ where $\bar{\mu} = \frac{1}{n} \sum_{i=1}^{n} \mu_i$, and set $r$ to be 3 for all experiments. For $\beta$, we consider the following scenarios:

(A) The first 10 regression coefficients are drawn uniformly between $-1$ to $1$, the rest are zero;

(B) The first 100 regression coefficients are drawn uniformly between $-0.5$ to $0.5$, the rest is zero;
Regression coefficients are drawn uniformly between $-0.1$ to $0.1$.

We normalize $\beta$ so that $\|\beta\| = 1$. The third scenario is referred to as the Dense case, and scenario A (Sparsity 1) is much sparser than scenario B (Sparsity 2). The sample size $n = 50$, the data dimension is $p = 10,000$, and $\rho = 0.5$ for the main experiments, and also consider other settings of $(n, p)$ and $\rho$ in other experiments for further understanding and comparison. Since the true $\mu_i$ is known in the simulation, we measure the performance with $\text{MSE} = \sum_{i=1}^{n} (\hat{\mu}_i - \mu_i)^2$ where $\hat{\mu}_i$ are the predictions from different methods.

**Performance of our method.** We evaluate the performance of ENSURE.

We set the ensemble number $T$ equal to 20 and the subset size is set $p_s = cn$, where $c$ is 5-fold cross-validated among $\{0.5, 1, 2, 4, 6, 10\}$ based on only one feature subset sampled from the full features. We plot the results for AR and T2 in Figure 1. From the plots, we have several observations. (a) The MSE decreases as the number of ensembles, $T$, grows. Combining ensemble, the subset regression even can outperform the regression based on full features remarkably. But the effect of increasing $T$ becomes weak after $T > 10$. It provides a guide that we can choose $T = 10$ in practice. (b) For the AR and T2 settings, the performance of ensemble subset prediction is much better than the methods based on full features, except for Dense of T2. Note that combining 10 predictions based on the subset of size $p_s = 500$ makes use of at most 5000 features, but it achieves much better performance than
Figure 1: The performance of the ensemble method for the AR and T2 settings: The MSE against ensemble number. From left to right: Sparsity 1, 2, and Dense. From top to bottom: AR and T2. The dashed lines correspond to ridge, LASSO, RF, or SIS.

methods using all features. (c) Comparing the Sparsity 1, 2 and Dense cases yields a useful observation. Figure I tells us that ENSURE has good performance across

21
all sparsity scenarios. It attains relatively better performance as the coefficients are sparser. This is a striking observation. When the coefficients are very sparse, the probability of getting important variables using subset regression is tiny. This may indicate that, when our aim is prediction, the identification of important variables may be not very necessary because of the curse of dimension and possible correlation among variables. Therefore, for high-dimensional prediction, ENSURE benefits from both a statistical and computationally-simple perspective.

Effect of the $p_s$ choice. We conduct to verify the performance of ensemble subset prediction under various $p_s \in \{50, 100, 200, 300, 500\}$, and report the performance in Figure 2. The size $p_s$ has some effect on the performance. The case of $p_s = 200$ performs best for Sparsity 1, Sparsity 2 and Dense. It indicates that the ensemble subset prediction may be not good to choose $p_s$ too large. More importantly, we check the effect of the $p_s$ choice by 5-fold CV. Generally, the CV choice does not attain the best performance, but in practice it obtain a good performance across all sparsity scenarios (Figure 1).

Effect of correlation of features. We next check how ENSURE performs when the correlation among features is different. We consider different $\rho$ in the AR setting. To study the effect of varying $\rho$, $\rho = 0, 0.8$ in Figure 4. From this Figure, we find that our approach’s good performance does not rely on the correlation among features, so that it is potentially suitable in a variety of applications.
Figure 2: The performance of the ensemble method for the AR setting. Various $p_s \in \{50, 100, 200, 300, 500\}$ and the $p_s$ choice by CV are considered.

**Effect of $(p, n)$**. We next consider different settings of $(p, n)$, specifically, $(p, n) = (1000, 20)$ and $(500, 20)$, and report the results in Figure 3. Our method works well under both settings. This results shows the robustness of our method to $(p, n)$. By comparing $(p, n) = (1000, 20)$ and $(p, n) = (500, 20)$, we see that our approach works better when $p$ is bigger with respect to $n$. It suggests that ENSURE may lose the advantage when $p$ is not much larger than $n$. It makes sense because the subset regression has a requirement on $p_s$ with respect to $n$ as shown in Theorem 2.

**Computational time comparison**. Finally, we compare computational costs. In Table 1 we report the computational cost of ENSURE under three settings $(n, p) = (20, 10K), (50, 10K), (50, 50K)$ and $(100, 100K)$. To show the computational advantage of ENSURE, we also report computation times for five other methods. Ridge regression is solved by using a QR decomposition. The LASSO is implement-
Figure 3: The performance of the ensemble method under various \((p, n)\). The input \(X\) is from AR under \(\rho = 0\) and is constructed by using the R package “glmnet” ([Friedman et al., 2010]), random forests by the R package “randomForest” ([Breiman, 2001]), and SIS by the R package “SIS” ([Fan and Lv, 2008]). All default parameters in these packages are kept. We perform the computation by R software in PC with a 3 GHz intel i7 processor, 8 GB memory.
Figure 4: The performance of the ensemble method for the AR under $\rho = 0$ and 0.8. and OS X operation system. From Table 1, we see that when $p$ is relatively large compared to $n$, then ENSURE requires much less computation time compared to methods based on full features. In particular, the computational burden of RF and SIS increases much faster as $n$ or $p$ gets large. In contrast, ENSURE greatly reduces the computational cost, because our method has the following features: (i)
subset regression and the CV procedure are calculated via conducting single ridge regression from the feature subset, (ii) simple random sampling costs little, and (iii) the proposed ensemble procedure is linear of the ensemble number. Based on these features, our method has a remarkable advantage in computation.

Table 1: Computational cost (seconds) including user and system time for ENSURE \((T = 1, 5, \text{ and } 10)\), and methods using all features: ridge, LASSO, RF, and SIS. The results are average values of 20 runs.

<table>
<thead>
<tr>
<th>((n, p))</th>
<th>time</th>
<th>(T) (ensure)</th>
<th>ridge</th>
<th>LASSO</th>
<th>RF</th>
<th>SIS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>5</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(20, 10K)</td>
<td>User time</td>
<td>0.04</td>
<td>0.05</td>
<td>0.06</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>System time</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.12</td>
<td>0.48</td>
<td>3.35</td>
<td>8.15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(50, 10K)</td>
<td>User time</td>
<td>0.17</td>
<td>0.27</td>
<td>0.33</td>
<td></td>
</tr>
<tr>
<td></td>
<td>System time</td>
<td>0.01</td>
<td>0.03</td>
<td>0.03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.27</td>
<td>0.79</td>
<td>15.0</td>
<td>19.43</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(50, 50K)</td>
<td>User time</td>
<td>0.22</td>
<td>0.31</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td>System time</td>
<td>0.03</td>
<td>0.07</td>
<td>0.07</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.45</td>
<td>3.99</td>
<td>91.68</td>
<td>91.02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(100, 100K)</td>
<td>User time</td>
<td>1.48</td>
<td>1.86</td>
<td>2.40</td>
<td></td>
</tr>
<tr>
<td></td>
<td>System time</td>
<td>0.34</td>
<td>0.35</td>
<td>0.39</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.04</td>
<td>14.87</td>
<td>529.4</td>
<td>284.5</td>
<td></td>
</tr>
</tbody>
</table>
5.2 Two Real Data Sets

We now analyze two real datasets. One dataset is gene microarray RMA (Scheetz et al., 2006) which consists of gene expression levels of 31,041 genes obtained from 120 rats, i.e., \((n,p) = (120,31041)\). The target variable is the TRIM32 gene. The other dataset is PUL (Ziyatdinov et al., 2015) from UCI machine learning repository. This dataset contains 58 time series acquired from 16 chemical sensors under a gas flow modulation. The sensors were exposed to gaseous binary mixtures of acetone and ethanol at different concentrations. We use the 1st chemical sensor, the response variable is acetone concentration. There are 7500 features, so \((n,p) = (58,7500)\).

Similar to the analysis on simulated data sets, we compare ENSURE with the methods based on full features. Since, of course we do not know the true parameters, we calculate prediction performance with test data. Both data sets were divided into two parts, training and testing data sets, by randomly selecting half-and-half observations. The results of 500 replicated experiments are summarized in Figure 5, where “MSPE” is the prediction error for the test dataset defined as 

\[ \text{MSPE} = E\|\hat{y} - y\|^2, \]

and the values are arithmetic means of 200 replicated experiments.

For RMA, ENSURE method attains the best one among the high-dimensional methods based on full features. For PUL, ENSURE obviously beats the methods of full features. These results are consistent with the observation of synthetic data sets. They show that ENSURE may be a better way to high-dimensional prediction than
methods based on full features.

![Graphs showing performance of ensemble methods for datasets RMA and PUL.]

Figure 5: The performance of the ensemble method of the data sets RMA and PUL.

6. Conclusion and Discussion

The analysis of high-dimensional data has attracted a lot of attention. The prediction performance and computational cost have become serious concern when the predictors’ dimension is large, which is not unusual. In this article, we have proposed a novel and easily implemented algorithm for high-dimensional prediction. We perform the subset regression based on a feature subset, which is chosen by sampling without replacement uniformly. Moreover, we apply the ensemble method into the subset regression. We have provided a theoretical justification for this procedure through an excess risk analysis. We have conducted intensive experiments to demonstrate that the procedure has promising performance.
There are three issues to address in the future. Firstly, it is still unclear why ENSURE does work in many empirical studies, such as the synthetic and real data sets investigated in this paper. Theorem 1 provides a bound without considering the estimation process of the coefficients, and Theorem 2 provides an upper bound on the excess risk by making use of the randomness from sampling. However, the mechanism behind these is an unaddressed problem. Understanding the mechanism can help us know when and where the subset regression has nice performance for the high-dimensional prediction.

Secondly, we construct our subset regression by $l_2$-regularized least-squares because of its theoretical convenience. Constructing alternative subset regression functions by using other high-dimensional methods, for example, $l_1$-regularization or other related methods, may deserve further studies.

Lastly, we use the simplest sampling, uniform sampling, due to the advantages of computational cost and theoretical convenience on approximation accuracy. However, from the approximation accuracy viewpoint, there may exist some more efficient sampling methods which more reply on data than uniform sampling, since the features of the datasets may be very heterogeneous. Therefore, developing an efficient data-drive sampling method is a worthy investigation.
Appendices

A.1. Proofs of the Upper Bound of the Subset Regression

Here we provide the excess risk bound under the general sampling framework rather than just uniform sampling; that is, we assume that the subset $S$ is drawn by sampling from features $\{1, \ldots, p\}$ proportional to the probabilities $\{\pi_1, \ldots, \pi_p\}$. Let $\Phi = \text{diag}\{\pi_k\}_{k=1}^p$, and $\Phi_s$ is the partition of $\Phi$ corresponding to the subset $S$. Note uniform sampling is a specific case, i.e., $\pi_k = 1/p$, so Theorem 2 will be proved by setting $\pi_k = 1/p$.

A.1.1 Two Lemmas

Lemma A.1. Let $M = XX^\top + \lambda I_n$ and $\Delta = XX^\top - X_s \Phi_s^{-1} X_s^\top$. Under the event that

$$\lambda_{\max}(M^{-1/2} \Delta M^{-1/2}) \leq \kappa$$

(A.1)

holds, we have

$$\|d_B\|^2 \leq (1 - \kappa)^{-2} \lambda^2 \|M^{-1}X\beta\|^2_2,$$

$$\|d_V\|^2 \leq (1 - \kappa)^{-2} \|M^{-1}XX^\top \epsilon\|^2.$$

Proof. Let $M_s = X_s \Phi_s^{-1} X_s^\top + \lambda I_n$. Firstly, we investigate the bias term. Since
\[ M^{-1/2} \Delta M^{-1/2} = I_n - M^{-1/2} M_s M^{-1/2}, \]

\[ M^{1/2} M_s^{-1} M^{1/2} = (I_n - M^{-1/2} \Delta M^{-1/2})^{-1}. \] \hspace{1cm} (A.2)

From (A.2),

\[ \|d_B\|^2 = \lambda^2 \|M_s^{-1} X \beta\|^2 = \lambda^2 \|M^{-1/2} M^{1/2} M_s^{-1} M^{1/2} M^{1/2} M^{-1} X \beta\|^2 \]

\[ \leq \lambda^2 \left[ \lambda_{\text{max}}(M^{1/2} M_s^{-1} M^{1/2}) \right]^2 \|M^{-1} X \beta\|^2 \]

\[ \leq \lambda^2 \left[ 1 - \lambda_{\text{max}}(M^{-1/2} \Delta M^{-1/2}) \right]^{-2} \|M^{-1} X \beta\|^2. \] \hspace{1cm} (A.3)

If \( \lambda_{\text{max}}(M^{-1/2} \Delta M^{-1/2}) \leq \kappa \), so from (A.3) we get

\[ \|d_B\|^2 \leq (1 - \kappa)^{-2} \lambda^2 \|M^{-1} X \beta\|^2. \]

Secondly, we investigate the variance term. Denote \( K = XX^T \). From (A.2), we thus have

\[ \|d_V\|^2 = \|M_s^{-1} (M_s - \lambda I) \epsilon\|^2 = \|M^{-1/2} M^{1/2} M_s^{-1} M^{1/2} M^{1/2} M^{-1/2} (M_s - \lambda I) \epsilon\|^2 \]

\[ \leq \left[ 1 - \lambda_{\text{max}}(M^{-1/2} \Delta M^{-1/2}) \right]^{-2} \left[ \lambda_{\text{max}}(K^{-1/2} (K - \Delta) K^{-1/2}) \right]^2 \|M^{-1} K \epsilon\|^2 \]

\[ \leq \left[ 1 - \lambda_{\text{max}}(M^{-1/2} \Delta M^{-1/2}) \right]^{-2} \|M^{-1} K \epsilon\|^2. \]

Thus, we have that, if \( \lambda_{\text{max}}(M^{-1/2} \Delta M^{-1/2}) \leq \kappa \),

\[ \|d_V\|^2 \leq (1 - \kappa)^{-2} \|M^{-1} K \epsilon\|^2. \]

\[ \square \]
Lemma A.2. Denote $d_1 \geq d_2 \geq \cdots \geq d_n > 0$ as the eigenvalues of $p^{-1}XX^T$. Let $A = \max\{(p\pi_k)^{-1}\}$
\[\|x^{(k)}\|^2 \}_{k=1}^p.\] We have, for all $t > 0$,
\[Pr\left(\lambda_{\max}(M^{-1/2}\Delta M^{-1/2}) > t\right) \leq n \exp\left\{\frac{-ps^2/2}{d^{-1}A + t/3}\right\}. \tag{A.4}\]

Proof. Motivated by \cite{Bach2013}, we study the probability bound under the sampling with replacement, then apply the theorem in \cite{Gross2010} to get the bound under the sampling without replacement. Let $\Delta_{\text{with}}$, which has the same expression as $\Delta$, be obtained by sampling independently $p$ features with replacement. We thus have
\[M^{-1/2}\Delta_{\text{with}}M^{-1/2} = \sum_{j=1}^{ps} \frac{1}{ps} \left(\sum_{k=1}^{p} x^{(k)}(x^{(k)})^\top - \sum_{k=1}^{p} \frac{I_{kj}}{\pi_k} x^{(k)}(x^{(k)})^\top\right) =: \sum_{j=1}^{ps} M_j,\]
where $I_{kj}$ is a random variable such that $\Pr(I_{kj} = 1) = \pi_i$ for the $k$-feature during the $j$th draw. From the process of sampling with replacement, random matrices series \[\{M_j\}_{j=1}^{ps}\] are independently distributed with $\mathbb{E}(M_j) = 0$ and
\[\mathbb{E}(M_j^2) = \frac{1}{ps} \left[\sum_{k=1}^{p} \frac{1}{\pi_k} \{x^{(k)}M^{-1}(x^{(k)})^\top\}^2 - (M^{-1/2}XX^T M^{-1/2})^2\right]. \tag{A.5}\]
Denote $d_1 \geq d_2 \geq \cdots \geq d_n > 0$ as the eigenvalues of $p^{-1}XX^T$. Let $A = \max\{(p\pi_k)^{-1}\}$. We have, for all $t > 0$,
max\{(p\pi_k)^{-1}\|x^{(k)}\|^2\}_{k=1}^P \cdot \lambda_{\text{max}}(\sum_{j=1}^{p_k} E(M_j^2)) = \frac{1}{p_s} \lambda_{\text{max}}\left(AM^{-1}XX^\top M^{-1} - (M^{-1/2}XX^\top M^{-1/2})^2\right) \\
\leq p_s^{-1} A \lambda_{\text{max}}(pM^{-1}XX^\top M^{-1}) = p_s^{-1} A \frac{d_n}{(d_n + p^{-1}\lambda)^2} \\
< p_s^{-1} Ad_n^{-1}. \quad (A.6)

On the other hand, from the matrix norm inequality

$$\lambda_{\text{max}}(M_j) \leq \frac{1}{p_s} \lambda_{\text{max}}\left(M^{-1/2}XX^\top M^{-1/2}\right) = \frac{1}{p_s} \frac{d_1}{d_1 + p^{-1}\lambda} < p_s^{-1}. \quad (A.7)$$

From (A.6) and (A.7), we apply the matrix Bernstein inequality of Tropp (2012) into \(\Delta_{\text{with}}\) to obtain its probability bound:

$$\Pr(M^{-1/2}\Delta_{\text{with}}M^{-1/2} > t) \leq n \exp\left(-\frac{p_s t^2/2}{(d_n^{-1} A + t/3)}\right). \quad (A.8)$$

By Gross and Nesme (2010), we have that for all \(g \in \mathbb{R}\)

$$\mathbb{E}[\text{tr}\{\exp(gM^{-1/2}\Delta M^{-1/2})\}] \leq \mathbb{E}[\text{tr}\{\exp(gM^{-1/2}\Delta_{\text{with}} M^{-1/2})\}]. \quad (A.9)$$

Combining (A.8) and (A.9) leads to the desired result.

A.1.2 Proof of Theorem 2

From the expression of \(\hat{\mu}_s - \mu\) in Eqn. (3.13) of Section 3, we have that

$$\mathbb{E}\|\hat{\mu}_s - \mu\|^2 = \mathbb{E}\|d_B\|^2 + \mathbb{E}\|d_V\|^2. \quad (A.10)$$
Write the event that for $0 < \kappa < 1$, 

$$\mathcal{E} = \{\lambda_{\max}(M^{-1/2}\Delta M^{-1/2}) < \kappa\}.$$ 

So if the event $\mathcal{E}$ holds, then from Lemma A.1

$$E\|\hat{\mu}_s - \mu\|^2 \leq (1 - \kappa)^{-2}(\lambda^2\|M^{-1}X\beta\|_2^2 + E\|M^{-1}XX^T\|_F^2)$$ 

$$= (1 - \kappa)^{-2}\left[\lambda^2\|(XX^T + \lambda I_n)^{-1}\mu\|^2 + \sigma^2\|I_n - \lambda(XX^T + \lambda I_n)^{-1}\|_F^2\right].$$

(A.11)

Notice that

$$\|(XX^T + \lambda I_n)^{-1}\mu\|^2 = \sum_{i=1}^{n} \frac{\mu_i^2}{(pd_i + \lambda)^2},$$

(A.12)

$$\|I_n - \lambda(XX^T + \lambda I_n)^{-1}\|_F^2 = \sum_{i=1}^{n} \frac{pd_i}{pd_i + \lambda},$$

(A.13)

On the other hand, from Lemma A.2 if $p_s > (\ln n / \rho)^{\kappa/3 + d_n^{-1}A}$, then

$$\Pr(\mathcal{E}) \geq 1 - \rho.$$ (A.14)

Combining (A.11), (A.12), (A.14) and Theorem 1, we have that

$$E\|\hat{\mu}_s - \mu\|^2 \leq (1 - \kappa)^{-2} \left[\lambda^2 \sum_{i=1}^{n} \frac{\mu_i^2}{(pd_i + \lambda)^2} + \sigma^2 \sum_{i=1}^{n} \frac{pd_i}{pd_i + \lambda}\right]$$ (A.15)

with probability at least $1 - \rho$. Therefore, Theorem 2 is proved.
A.2. Proof of Theorem 3

From the process of the subset-ensemble prediction, we have that

$$\|\hat{\mu}_{\text{ensure}} - \mu\| = \|\frac{1}{T} \sum_{t=1}^{T} \hat{\mu}_{s,t} - \mu\| = \|\frac{1}{T} \sum_{t=1}^{T} (\hat{\mu}_{s,t} - \mu)\|.$$  

It follows that

$$E\|\hat{\mu}_{\text{ensure}} - \mu\|^2 = E\|\frac{1}{T} \sum_{t=1}^{T} (\hat{\mu}_{s,t} - \mu)\|^2$$

$$= \frac{1}{T^2} \left( \sum_{t_1 \neq t_2} E(\hat{\mu}_{s,t_1} - \mu)\top (\hat{\mu}_{s,t_2} - \mu) + \sum_{t=1}^{T} E\|\hat{\mu}_{s,t} - \mu\|^2 \right)$$

$$= \frac{1}{T^2} \left\{ \sum_{t_1 \neq t_2} E_m \left( [E(\hat{\mu}_{s,t_1} - \mu|y)]\top [E(\hat{\mu}_{s,t_2} - \mu|y)] \right) + \sum_{t=1}^{T} E\|\hat{\mu}_{s,t} - \mu\|^2 \right\}$$

$$= (1 - \frac{1}{T}) \left( \|E(d_B)\|^2 + E_m \|E(d_V|y)\|^2 \right) + \frac{1}{T} (E\|d_B\|^2 + E\|d_V\|^2)$$  

(A.16)

where the 3rd equality is from that each feature subset is independently and repeatedly in our ensemble process, $\hat{\mu}_{s,t} : t = 1, \ldots, T$ can be considered as i.i.d. random vectors.

$$E\|\hat{\mu}_s - \mu\|^2 - E\|\hat{\mu}_{\text{ensure}} - \mu\|^2$$

$$= (1 - \frac{1}{T}) \left\{ E\|d_B\|^2 + E\|d_V\|^2 - \|E(d_B)\|^2 - E\|E(d_V|y)\|^2 \right\}$$

$$= (1 - \frac{1}{T}) \left[ E\|d_B\|^2 - \|E(d_B)\|^2 + E_m \{ E(\|d_V\|^2|y) \} - E\|E(d_V|y)\|^2 \right]$$

$$= (1 - \frac{1}{T}) \{ \sigma_B^2 + E_m(\sigma_V^2) \}.$$  

Thus, Theorem 3 is proved.
Acknowledgements

The authors thank two reviewers for their insightful comments and suggestions that significantly improve the paper. Zhu’s work was partially supported by National Natural Science Foundation of China grants 11301514 and 71532013, and by Shanghai Municipal Science and Technology Major Project (No.2018SHZDZX01) and 111 Project (No.B18015).

References


Candes, E. and T. Tao (2007). The Dantzig selector: statistical estimation when \( p \) is much larger than \( n \).


neural information.


Institute of Science and Technology for Brain-inspired Intelligence

Key Laboratory of Computational Neuroscience and Brain-Inspired Intelligence

MOE Frontiers Center for Brain Science

Fudan University

Shanghai 200433, China

E-mail: (rongzhu@fudan.edu.cn)

Department of Statistics

George Washington University

Washington, DC 20052, USA

E-mail: (hliang@gwu.edu)

Department of Statistics and Data Science

School of Operations Research and Information Engineering

Cornell University, Ithaca

New York 14853, USA

E-mail: (dr24@cornell.edu)