Balance-Subsampled Stable Prediction across Unknown Test Data

KUN KUANG†, Zhejiang University
HENGTAO ZHANG†, The University of Hong Kong
RUNZE WU, Fuxi AI Lab, NetEase Games
FEI WU, Zhejiang University
YUETING ZHUANG, Zhejiang University
AIJUN ZHANG*, The University of Hong Kong

In data mining and machine learning, it is commonly assumed that training and test data share the same population distribution. However, this assumption is often violated in practice because of the sample selection bias, which might induce the distribution shift from training data to test data. Such a model-agnostic distribution shift usually leads to prediction instability across unknown test data. This paper proposes a novel balance-subsampled stable prediction (BSSP) algorithm based on the theory of fractional factorial design. It isolates the clear effect of each predictor from the confounding variables. A design-theoretic analysis shows that the proposed method can reduce the confounding effects among predictors induced by the distribution shift, improving both the accuracy of parameter estimation and the stability of prediction across unknown test data. Numerical experiments on synthetic and real-world data sets demonstrate that our BSSP algorithm can significantly outperform the baseline methods for stable prediction across unknown test data.

CCS Concepts: •**Computing methodologies** →**Causal reasoning and diagnostics;** *Machine learning; Statistical relational learning;*

Additional Key Words and Phrases: Stable Prediction, Distribution Shift, Sub-sampling, Variable Deconfounding, Fraction Factorial Design.

ACM Reference format:
Kun Kuang†, Hengtao Zhang†, Runze Wu, Fei Wu, Yueting Zhuang, and Aijun Zhang*. 2021. Balance-Subsampled Stable Prediction across Unknown Test Data. *ACM Trans. Knowl. Discov. Data.* 1, 1, Article 1 (January 2021), 23 pages.
DOI: 10.1145/3477052

1 INTRODUCTION

One of the most common assumptions in the machine learning algorithms is that the training data consists of samples drawn randomly from the same underlying distribution as the test samples. Under this assumption, many machine learning and artificial intelligence algorithms have been proposed and shown to be successful in many fields, such as nature

†Kun Kuang and Hengtao Zhang contribute equally to this paper. *Aijun Zhang is the corresponding author (Email: ajzhang@umich.edu).

DOI: 10.1145/3477052

language process [57], computer vision [21], healthcare [43]. However, in many practical applications, this assumption is often violated because of sample selection bias, which brings distribution shifts between observed data and the population. It would cause the distribution shift between training and test data. Moreover, the test data is often unobserved during model training, which leads to the agnostic distribution shift problem. Therefore, it is highly demanding to develop predictive algorithms that are stable/robust to the agnostic distribution shift between training and unknown test data.

Recently, many invariant learning algorithms have been proposed to address the agnostic distribution shift problem, including domain generalization [38], invariant causal prediction [40] and causal transfer learning [42]. These methods explore the invariant representation of data, the invariant structure between predictors and outcome variables, and causal structure across multiple training datasets. However, their performance usually heavily depends on the diversity of multiple training datasets so that the distribution shift that does not appear in existing datasets can not be appropriately addressed. Moreover, their training complexity grows exponentially with the dimension of the feature space in the worst case, which is not acceptable in practice.

To address the problem of stable prediction, in this paper, we assume that the underlying predictive mechanism between predictors/features $\mathbf{X}$ and outcome variable Y is invariant across datasets. Based on the invariant predictive mechanism, all predictors $\mathbf{X}$ fall into one of two categories. One category includes stable features $\mathbf{S}$, which have causal effects on outcome Y, and are stable/invariant across datasets. For example in computer vision, ears, noses, and legs of dogs are stable features to recognize whether an image contains a dog or not. The other category includes noisy features $\mathbf{V}$, which have no causal effects on outcomes, but might be highly correlated with either stable features, the outcome variable or both in certain datasets. For the same example, the grass and background pixels are noisy features for dog recognition. Hence, taking the regression task as an example, we set $\mathbf{X} = \{\mathbf{S}, \mathbf{V}\}$ and have $Y = f(\mathbf{X}) + \epsilon = f(\mathbf{S}) + \epsilon$ in our problem. Conditional on the full set of stable features, the noisy features do not affect the expected outcome. However, the distribution shift might make a part of noisy features become power predictors. In the previous example, grass would be a power predictor if most of the dogs in the training data are on the grass. Therefore, the distribution shift leads to potential confounding and spurious correlation between the noisy features and the outcome variable[1]. To address the stable prediction problem, we should reduce such confounding effects and spurious correlations between the noisy features and the outcome variable.

In practice, we have no prior knowledge on which features are stable and which are noisy. Under such a setting, one possible way to remove the spurious correlation is to isolate the impact of each individual feature on the response. Variable balancing techniques are widely used for causation recovery in the literature of causal inference [28]. The key idea is to construct sample weights by either employing propensity scores [3, 25, 27, 44] or optimizing weights directly [2, 19, 26, 61]. Recently, a global balancing algorithm [24] was proposed to learn the weights that enforce all features to be as independent as possible, which was shown to have better performance. However, this algorithm only focuses on the pairwise confounding effects, while ignoring the higher-order interactions. Moreover, it is not an efficient way to learn the weight for each sample and use the full data to perform model training in the big data scenario.

[1]Here, the calculated correlation between the noisy features and the outcome variable is called spurious correlation, since the generation of outcome variable does not depend on the noisy features.

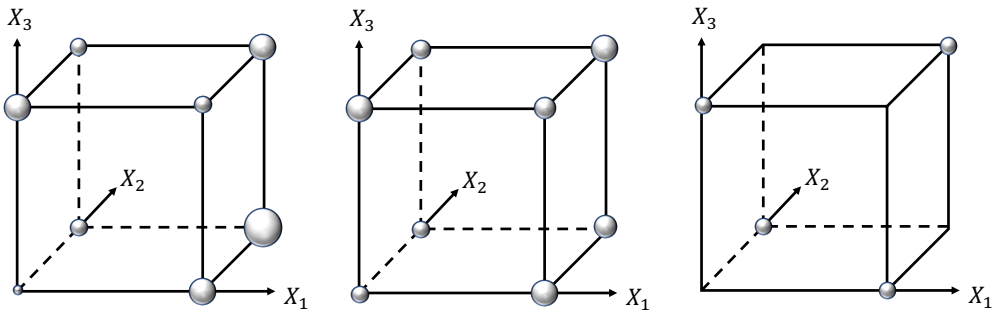


Fig. 1. A toy example to illustrate the main idea of each deconfounding method.

Full and fractional factorial designs are widely used in statistics for arranging factorial experiments without confounding effects [8, 11]. With data collected from a factorial designed experiment, one can easily isolate the impact of each feature and reveal the causation between predictors and the outcome variable. Inspired by the factorial designs of experiments, we propose a Balance-Subsampled Stable Prediction (BSSP) algorithm, which consists of a factorial design-based subsampling strategy for covariates balancing and a subsampled learning model for stable prediction. Using the factorial design, the subsampling strategy selects a subset of samples from training data such that the covariates are mutually balanced and thus deconfounded. Thus, the model fitted by the subsamples would exploit the causations between predictors and outcome for stable prediction.

Our BSSP method can be regarded as a general data pretreatment method that reduces the confounding among predictors for prediction. This approach does not impose restrictions on the choice of models and the type of responses. The samples selected by the proposed method may work for generalized linear models such as linear regression and logistic regression, and more complex models like neural networks. The response variable can be either continuous or binary. We demonstrate the advantages of the proposed BSSP algorithm for stable prediction on both regression and classification tasks based on synthetic and real-world datasets. It is shown to have overwhelming performance across unknown test data with a distribution shift from the training data, thus achieving a more stable prediction. Furthermore, we can train the model faster as the subdata is much smaller than the full data.

To our best knowledge, this paper is the first to establish a connection between the fractional factorial design and the stable learning problem. The main idea of different deconfouding methods is illustrated with a toy example in Fig. 1. Consider a three-dimensional dataset with binary inputs. We visualize the sample space in Fig. 1a, where the bubble size corresponds to the number of observations on that point. Each facet corresponds to the conditional distribution of two variables w.r.t. the remaining one. In the ideal case, all bubbles should have the same size and there are no confounding effects among variables, since any two opposite facets have the same distribution. To achieve this goal, the global balancing method (Fig. 1b) takes all sample values but reweighs them to change the data distribution. Based on the fractional factorial design, our subsampling method (Fig. 1c) only uses a fraction of samples that approximate the ideal situation. For example, the

conditional distributions of $X_i|X_k$ and $X_j|X_k$ are the same in Fig. 1c, where $i, j, k \in \{1, 2, 3\}$ are distinctive indices.

In summary, the contributions of this paper include:

- We investigate the problem of stable prediction across unknown test data, where the distribution of agnostic test data is different from the training data.
- We propose a novel BSSP algorithm based on the fractional factorial design strategy for variable deconfounding and stable prediction. For the first time the subsampling technique is introduced to the study of stable prediction problem.
- Theoretically, we show that the fractional factorial design-based subsampling can remove the confounding effects with non-linear interactions. Hence, our BSSP algorithm can precisely estimate the parameters and achieve a stable prediction across unknown environments.
- We conduct extensive experiments on synthetic and real-world datasets, and demonstrate the advantages of our algorithm for stable prediction in both regression and classification tasks.

2 RELATED WORK

To remedy the problem of distribution shift, a considerable effort has been made in domain adaptation [4, 5, 9] and transfer learning [31, 39]. Given the test data, domain adaptation accommodates a learning algorithm trained by the training data, such that the predictive error on the test data is minimized. These methods achieve good performance for correcting distribution shift in real applications. In natural language processing (as a new trend of learning in computational intelligence[22]), [7] proposed the structural correspondence learning from the feature perspective for the domain adaptation, which is further extended by [58–60] using the neural networks. [23] introduced instance weighting to adapt different domains. It also draws our attention that there are a series of works that leverage the genetic programming in document classification or sentiment analysis across different domains [10, 16, 17, 32], and [36] incorporated common semantic information into aspect-based sentiment analysis. The main drawback in these works is that one needs prior knowledge of the test distribution during the training, and needs to re-train the model for different test data. In this paper, we focus on the problem of stable prediction with agnostic distribution shift.

Recently, the invariant learning algorithms have been proposed to address the agnostic distribution shift problem, including domain generalization [38], invariant causal prediction [40], causal transfer learning [42] and invariant risk minimization [1]. These methods explore the invariant representation of data, the invariant structure between predictors and outcome variables, and causal structure across multiple training datasets. However, their performance depends heavily on the diversity of multiple training datasets, while the distribution shift that does not appear in existing datasets can not be appropriately addressed. Moreover, their training complexity grows exponentially with the dimension of the feature space in the worst case, which is not acceptable in practice.

To enhance the stability and robustness of artificial intelligence, recently, many methods have been proposed from different aspects, such as adversarial learning [41, 47], artificial general intelligent [33], distributional robustness optimization [13, 30], invariant/hetogeneous risk minimization [1, 29]. In this paper, we focus on the stability of model predictions on agnostic test data, whose distribution might be different with the training one.

Variables balancing is a key technique for treatment effect estimation in observational studies. [44] proposed to achieve variables balancing by sample reweighting with the inverse of the propensity score. [2] proposed an approximate residual balancing algorithm by combining outcome modeling with variables balancing. [26] jointly optimized sample weights and variable weights for a differentiated variable balancing. [14] studied the variables balancing for continuous treatment variable under linear assumption. These methods perform well on causal effect estimation in observational studies, but they are not designed for the case with many causal variables, such that they cannot immediately extend to our stable prediction problem.

Our work is closely related to [24], which proposed a global balancing algorithm for stable prediction. As shown in Eq. (7), the global balancing algorithm attempts to learn global sample weights for each sample such that all predictors may become independent. [24] also proved that the ideal global sample weights could isolate the impact of each predictor, hence address the stable prediction problem. However, the algorithm in [24] is non-convex and only focuses on the first-order confounding between any two variables, ignoring the higher-order interactions.

In statistical designs of experiments, full and fractional factorial designs, especially the two-level factorial designs, are widely used for experimental planning and data collection; see [11] and references therein. Resolution and minimum aberration are two main criteria to evaluate the goodness of a fractional factorial design; see [15, 34, 54, 55]. The factorial designs provide efficient ways of conducting experiments, but not for sample selection based on observational data.

Subsampling is an efficient strategy to accelerate the machine learning algorithms. Traditionally, the idea of sampling is a key concept in statistical surveys and estimation of point statistics [46]. Recently, subsampling appears to be an effective strategy for big data modeling, including the randomized leveraging methods [12, 35] and deterministic subsampling methods [50, 51, 56]. These subsampling methods aim to provide a fast approximation to the model parameters estimated by the full data. Unlike them, this paper considers the idea of subsampling for the stable prediction across different and possibly unknown datasets.

3 PROBLEM AND NOTATIONS

For a prediction problem, let $\mathbf{X}$ and Y denote the predictors and outcome variable, respectively. Define an **environment** to be the joint distribution P_{XY} of $\{\mathbf{X}, Y\}$. Let $\mathcal{E}$ denote the set of all environments, and $\mathbf{M}^e = \{\mathbf{X}^e, Y^e\}$ be the dataset collected from $e \in \mathcal{E}$. In real applications, the joint distribution of features and outcome can vary across environments: $P^e_{XY} \neq P^{e'}_{XY}$ for $e, e' \in \mathcal{E}$. Then the stable prediction problem [24] is defined as follows.

Problem 1 (Stable Prediction). *Given one training environment $e \in \mathcal{E}$ with dataset $\mathbf{M}^e = \{\mathbf{X}^e, Y^e\}$, the task is to **learn** a predictive model that can **stably** predict across unknown test environments $\mathcal{E}$.*

In this paper we consider the same problem setting of stable learning as in [24, 45, 52], where the features $\mathbf{X} \in \{0, 1\}^d$ are binary. The continuous variables can be converted to be binary via binning or direct dichotomization, while the discrete variables can be grouped into two categories[2].

[2]As an extended work, we are investigating multi-category and continuous features based on the similar idea of leveraging the multi-level factorial design and uniform experimental design.

Here, we measure the performance of stable prediction by *Average_Error* and *Stability_Error* [24] with the following definition:

$$Average\_Error = \frac{1}{|\mathcal{E}|}\sum_{e\in\mathcal{E}} Error(\mathbf{M}^e), \tag{1}$$

$$Stability\_Error = \sqrt{\frac{1}{|\mathcal{E}|-1}\sum_{e\in\mathcal{E}} (Error(\mathbf{M}^e) - Average\_Error)^2}, \tag{2}$$

where $Error(\mathbf{M}^e)$ represents the predictive error on dataset $\mathbf{M}^e$. Actually, the *Average_Error* and *Stability_Error* correspond to the mean and variance of the predictive error over all possible unknown test environments $e \in \mathcal{E}$.

Now let $\mathbf{X} = \{\mathbf{S}, \mathbf{V}\}$, where $\mathbf{S}$ denotes stable features and $\mathbf{V}$ denotes noisy features with following the assumption [24]:

Assumption 1. *There exists a probability function $P(y|s)$ such that for all environment $e \in \mathcal{E}$,*

$$P(Y^e = y|\mathbf{S}^e = s, \mathbf{V}^e = v) = P(Y^e = y|\mathbf{S}^e = s) = P(y|s). \tag{3}$$

Assumption 1 means that the responses are solely determined by the stable features. One can achieve stable prediction by developing a predictive model that learns the stable function $f(\mathbf{S})$ induced by $P(Y|\mathbf{S})$. For example, we have $f(\mathbf{S}) = \mathbb{E}(Y|\mathbf{S}) = \int YP(Y|\mathbf{S})dY$ when $Y = f(\mathbf{S}) + \varepsilon$ with the zero mean error ε. However in practice, we have no prior knowledge on which features belong to $\mathbf{S}$ or $\mathbf{V}$. Indeed, it is difficult to identify $\mathbf{V}$ as it typically demonstrates suspicious correlation with responses due to distribution shift.

In this paper, we study the stable prediction problem under model misspecification for continuous and binary responses (i.e., for both regression and classification tasks). Suppose that the true stable function $f(\mathbf{S})$ and Y in environment e are given by:

$$Y^e = f(\mathbf{S}^e) + \mathbf{V}^e\beta_V + \varepsilon^e = \mathbf{S}^e\beta_S + g(\mathbf{S}^e) + \mathbf{V}^e\beta_V + \varepsilon^e, \quad \text{for regression;} \tag{4}$$

$$\text{logit}(P(Y^e = 1|\mathbf{S}^e)) = \mathbf{S}^e\beta_S + g(\mathbf{S}^e) + \mathbf{V}^e\beta_V, \quad \text{for classification,} \tag{5}$$

where $\beta_V = \mathbf{0}$ and $\varepsilon^e \perp \mathbf{X}^e$. For simplicity, we restrict our attention on the regression case (4) in this section, and the classification scenario can be similarly derived. We assume that the analyst mis-specifies the underlying model (4) by omitting non-linear term $g(\mathbf{S}^e)$ and uses a linear model for prediction. Then, standard linear regression may estimate non-zero effects of noisy features $\mathbf{V}^e$ if they are correlated with the omitted term $g(\mathbf{S}^e)$ in the training environment e, which leads to instability on prediction since the following theorem implies that the correlation between $\mathbf{V}$ and $g(\mathbf{S})$ is changeable across unknown test environments.

Theorem 3.1. *Under assumption 1, the distribution shift across environments is induced by the variation in the joint distribution over* $(\mathbf{V}, \mathbf{S})$.

Proof. $P(\mathbf{X}^e, Y^e)$ can be decomposed as:

$$\begin{aligned} P(\mathbf{X}^e, Y^e) &= P(Y^e|\mathbf{X}^e)P(\mathbf{X}^e) \\ &= P(Y^e|\mathbf{S}^e, \mathbf{V}^e)P(\mathbf{S}^e, \mathbf{V}^e) \\ &= P(Y^e|\mathbf{S}^e)P(\mathbf{S}^e, \mathbf{V}^e) \end{aligned} \tag{6}$$

With assumption 1, we know the distribution $P(Y^e|\mathbf{S}^e) = P(Y^{e'}|\mathbf{S}^{e'})$ for different $e, e' \in \mathcal{E}$. Hence, the distribution shift across environments (i.e., $P(\mathbf{X}^e, Y^e) \neq P(\mathbf{X}^{e'}, Y^{e'})$) is induced by the variation in the joint distribution over $(\mathbf{V}, \mathbf{S})$ (i.e., $P(\mathbf{S}^e, \mathbf{V}^e) \neq P(\mathbf{S}^{e'}, \mathbf{V}^{e'})$). □

Notations. Let n be the sample size, and d be the dimensionality of variables. For any vector $\mathbf{v} \in \mathbb{R}^{d\times 1}$, let $\|\mathbf{v}\|_1 = \sum_{i=1}^{d} |v_i|$. For any matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$, let $\mathbf{X}_{i,\cdot}$ and $\mathbf{X}_{\cdot,j}$ represent the i^{th} sample and the j^{th} variable in $\mathbf{X}$, respectively. To simplify notations, we remove the environment variable e from $\mathbf{X}^e$, $\mathbf{S}^e$, $\mathbf{V}^e$, ε^e, and Y^e when there is no confusion from the context. For an integer $k \geq 1$, we use $[k]$ to denote the set of integer indices up to k, i.e., $[k] = \{1, \ldots, k\}$.

4 VARIABLE DECONFOUNDING

In this section, we first propose a generalized global balancing loss for the stable prediction. Then, we introduce the variable deconfounding technique based on the Factional Factorial Designs (FFDs) to optimize this generalized loss.

4.1 Generalized Global Balancing Loss

Theorem 3.1 implies that if the covariates are mutually independent (or there are no confounding effects among variables), we can well estimate parameter β_V in Eq. (4), hence improve the stability of prediction across unknown test environments. The confounding effects between covariates and the binary treatment status are typically eliminated by balancing covariates in causality literature [2, 19]. Recently, [24] successively regarded each variable as the treatment indicator and minimized a global balancing loss:

$$\begin{aligned}\min_{\mathbf{W}\in\mathbb{R}^n} \mathcal{L}(\mathbf{W}, \mathbf{X}) &= \sum_{j=1}^{d} \left\| \frac{\mathbf{X}_{\cdot,-j}^T \cdot (\mathbf{W} \odot \mathbf{X}_{\cdot,j})}{\mathbf{W}^T \cdot \mathbf{X}_{\cdot,j}} - \frac{\mathbf{X}_{\cdot,-j}^T \cdot (\mathbf{W} \odot (\mathbf{1} - \mathbf{X}_{\cdot,j}))}{\mathbf{W}^T \cdot (\mathbf{1} - \mathbf{X}_{\cdot,j})} \right\|_2^2 \\ &= \sum_{j=1}^{d} \sum_{k\neq j} \left[\frac{\sum_{i:X_{ij}=1} W_i X_{ik}}{\sum_{i:X_{ij}=1} W_i} - \frac{\sum_{i:X_{ij}=0} W_i X_{ik}}{\sum_{i:X_{ij}=0} W_i} \right]^2, \end{aligned} \tag{7}$$

where $\odot$ refers to Hadamard product, and $\mathbf{X}_{\cdot,-j} = \mathbf{X} \backslash \{\mathbf{X}_{\cdot,j}\}$ collects the remaining variables after removing the j^{th} variable. The difference in quadratic loss enforces $P_{\mathbf{W}}(X_k = 1|X_j = 1) \approx P_{\mathbf{W}}(X_k = 1|X_j = 0)$ w.r.t. the weighted conditional distribution $P_{\mathbf{W}}$. When the equation holds exactly, it can be shown that $X_k \in \{0, 1\}$ and $X_j \in \{0, 1\}$ are independent and thus have no confounding effects. $\mathcal{L}(\mathbf{W}, \mathbf{X})$ hence globally balances each variable with others by reweighting the observations.

There are several drawbacks for Eq. (7). Firstly, it can only remove the first-order confounding effects, but ignore higher-order ones[3] that between $\mathbf{V}$ and k-way interaction function $g(\mathbf{S})$, e.g., the two-way interaction $g(\mathbf{S}) = \mathbf{S}_{\cdot,1}\mathbf{S}_{\cdot,2}$. Moreover, it is very hard to find the global optimal solution as Eq. (7) is non-convex. Lastly, using full weighted data can be computationally expensive for especially big data scenarios.

[3]Here, we denote high-order confounding effect as the confounding effect between $\mathbf{V}$ and high-order function $g(\mathbf{S})$. In practice, the function $g(\mathbf{S}^e)$ may include not only the linear combinations of $\mathbf{S}^e$ but also some high-order interactions, such as the two-way interaction $\mathbf{S}_{\cdot,1}\mathbf{S}_{\cdot,2}$.

Considering high-order confounding effects among variables, we define a new *generalized global balancing* loss $\mathcal{L}_k(\mathbf{W}, \mathbf{X})$ as:

$$\mathcal{L}_k(\mathbf{W}, \mathbf{X}) = \sum_{j\in[d]} \sum_{I_k\subseteq[d]\setminus\{j\}} \left[\frac{\mathbf{X}_{I_k}^T \cdot (\mathbf{W} \odot \mathbf{X}_{\cdot,j})}{\mathbf{W}^T \cdot \mathbf{X}_{\cdot,j}} - \frac{\mathbf{X}_{I_k}^T \cdot (\mathbf{W} \odot (\mathbf{1} - \mathbf{X}_{\cdot,j}))}{\mathbf{W}^T \cdot (\mathbf{1} - \mathbf{X}_{\cdot,j})} \right]^2$$
$$= \sum_{j\in[d]} \sum_{I_k\subseteq[d]\setminus\{j\}} \left[\frac{\sum_{i:X_{ij}=1} W_i X_{iI_k}}{\sum_{i:X_{ij}=1} W_i} - \frac{\sum_{i:X_{ij}=0} W_i X_{iI_k}}{\sum_{i:X_{ij}=0} W_i} \right]^2, \tag{8}$$

where k refers to the order of confounding effect with $1 \le k < d$, and X_{iI_k} as well as $\mathbf{X}_{I_k}$ denotes the k-way interaction w.r.t. the index subset I_k. This loss broadly measures different orders of correlation or confounding effect between $\mathbf{V}$ and $g(\mathbf{S})$. It is easy to see that Eq. (7) is a special case of Eq. (8) with $k = 1$. Our target in this paper is to minimize the aggregation of $L_k(\mathbf{W}, \mathbf{X})$ up to the order k.

4.2 Variables Deconfounding via Fractional Factorial Designs

In this section, we elaborate on how fractional factorial designs can be used to deconfound the variables in terms of minimizing the generalized balancing loss in Eq. (8). Note that the binary-encoded data matrix is closely related to a two-level factorial design, which motivates us to leverage the classical results from the fractional factorial design literature.

Two-level fractional factorial design (FFD) [11]: It is a size-m subset of the full factorial design that consists of all 2^d possible combinations of the vector $\{-1, 1\}^d$. We denote FFD by $\mathbf{D} \in \{-1, 1\}^{m\times d}$, where $0 < m \le 2^d$.

One important feature of FFD is that variables and their interactions are orthogonal to some degrees, and they can achieve joint orthogonality when FFD becomes full factorial. Another cardinal observation is that the mean differences in Eq. (8) can be transferred into the inner products of the main effects and high-order interactions of a design in $\{-1, 1\}^{m\times d}$; see the proof of Theorem 4.3. Consequently, the orthogonality of FFD can help remove non-zero inner products and lead to a minimal loss.

Resolution [15], denoted as R, is an important criterion to reflect the degree of orthogonality. For an FFD, define the *generalized word-length pattern* [34]

$$W(\mathbf{D}) = (A_1(\mathbf{D}), \ldots, A_d(\mathbf{D})), \tag{9}$$

where $A_j(\mathbf{D})$ refers to the generalized wordlength and measures the degree of j-factor non-orthogonality. Specifically, we have for $j = 1, \ldots, d$,

$$A_j(\mathbf{D}) = \frac{1}{m(q-1)} \sum_{k=0}^{d} P_j(k; d, q) B_j(\mathbf{D}), \tag{10}$$

where q denotes the number of levels,

$$P_j(x; d, q) = \sum_{w=0}^{j} (-1)^w (q-1)^{j-w} \binom{x}{w} \binom{d-x}{j-w} \tag{11}$$

are the Krawtchouk polynomials [37], and $B(\mathbf{D}) = (B_0(\mathbf{D}), \ldots, B_d(\mathbf{D}))$ is the distance distribution $B_j(\mathbf{D}) = m^{-1}|\{(\mathbf{c}, \mathbf{d}) : d_H(\mathbf{c}, \mathbf{d}) = j, \mathbf{c}, \mathbf{d} \in \mathbf{D}\}|$ with $d_H(\cdot, \cdot)$ denoting the Hamming distance. Note that $B_j(\mathbf{D})$ is invariant to the encoding way of $\mathbf{D}$. Then, the resolution of $\mathbf{D}$ is defined as the smallest index $R \le d$ such that $A_R(\mathbf{D}) > 0$. Note that the full factorial design with $m = 2^d$ has resolution $d + 1$, since $A_j(\mathbf{D}) = 0$ for all $j \in [d]$.

The following lemma [20] explains the relationship between resolution R and orthogonal strength. We omit 'fractional factorial' in the resolution-R design without ambiguity to the context.

LEMMA 4.1. *The resolution-R design* $\mathbf{D}$ *has orthogonal strength* $t = R-1$*, where* t *means that one can see all possible* t*-tuples equally often for* $m/2^t$ *times in any* t *columns of* $\mathbf{D}$*.*

Lemma 4.1 implies that any t columns/variables of $\mathbf{D}$ contain $m/2^t$ full factorials such that every t factors are jointly orthogonal. Furthermore, this lemma also implies low order $t' < t$ orthogonal strength exists for the resolution-R design. In other words, FFD can preserve the joint orthogonality up to its resolution minus one. For example, a resolution-3 design guarantees the pairwise orthogonality among the main effects of all factors. The following theorem further states the preserved orthogonality among the main effect and their k-way interaction.

THEOREM 4.2. *Let* $I_k \subseteq [d]$ *denote any collection of distinctive factors with* $|I_k| = k \leq d$*,* $\mathbf{D} = (\mathbf{D}_{\cdot 1}, \ldots, \mathbf{D}_{\cdot d}) \in \{-1, 1\}^{m\times d}$ *be the design matrix, and* $\mathbf{D}_{I_k} \in \{-1, 1\}^m$ *represent* k*-way interaction of* I_k*. We have a)* $\mathbf{D}_{\cdot i}^T \mathbf{D}_{\cdot j} = 0, i \neq j$*, for any resolution-R design with* $R \geq 3$*; and b)* $\mathbf{D}_{I_k}^T \mathbf{D}_{\cdot j} = 0, j \in [d], 2 \leq k \leq R-2$*, for any resolution-R design with* $R \geq 4$*.*

PROOF. To prove above theorem, we first inductively show a lemma that any full factorial design (FD) denoted by $\mathbf{D} \in \{-1, 1\}^{2^d \times d}$ has $\mathbf{1}^T \mathbf{D}_{I_d} = 0$ for the integer $d \geq 1$. It is easy to check that $\mathbf{1}^T \mathbf{D}_{I_d} = 0$ holds when $d = 1, 2$. Suppose this equality holds for any integer $d = \ell, \ell \geq 1$. When $d = \ell + 1$, note that $\mathbf{D}$ is invariant to the permutation of rows, so we rearrange the first column and have

$$\mathbf{D}_{I_{\ell+1}} = (\underbrace{-1, \ldots, -1}_{2^\ell}, \underbrace{1, \ldots, 1}_{2^\ell})^T \odot (\mathbf{D}_{I_\ell}^{(1)}, \mathbf{D}_{I_\ell}^{(2)})^T, \tag{12}$$

where the sub-designs $\mathbf{D}^{(1)}, \mathbf{D}^{(2)} \in \{-1, 1\}^{2^\ell \times \ell}$ also belong to FD [20]. Therefore, it can be derived that $\mathbf{1}^T \mathbf{D}_{I_{\ell+1}} = \mathbf{1}^T \mathbf{D}_{I_\ell}^{(2)} - \mathbf{1}^T \mathbf{D}_{I_\ell}^{(1)} = 0$ and the statement gets proved. So for any FFD with orthogonal strength t, we have

$$\mathbf{1}^T \mathbf{D}_{I_k} = 0 \quad \text{for} \quad k \in [t], \tag{13}$$

because Lemma 1 tells that all combinations of at most t-tuples (full factorial design) appear with equal frequency in corresponding distinctive columns.

With the above property, we can easily show the first case in the theorem as $\mathbf{D}_{\cdot i}^T \mathbf{D}_{\cdot j} = \mathbf{1}^T \mathbf{D}_{I_2}$ with $I_2 = \{i, j\}$, and the resolution-R design has orthogonal strength $t = R - 1 \geq 2$, which implies $\mathbf{D}_{\cdot i}^T \mathbf{D}_{\cdot j} = 0$. For the second case, we restate it in terms of orthogonal strength t, that is, we need to show $\mathbf{D}_{I_k}^T \mathbf{D}_{\cdot j} = 0$, $j \in [d], 2 \leq k \leq t-1$ for $t \geq 3$, which can be inductively proved in the similar manner. Without loss of the generality, we just show $\mathbf{D}_{I_k}^T \mathbf{D}_{\cdot j} = 0$ for $t = 3$ in what follows. When $j \notin I_2$, we can construct a 3-column FFD with indices $I_3 = \{j\} \cup I_2$ and $\mathbf{D}_{I_2}^T \mathbf{D}_{\cdot j} = \mathbf{1}^T \mathbf{D}_{I_3}$. And we can similarly obtain $\mathbf{1}^T \mathbf{D}_{I_3} = 0$ as done in the first case because of $t = 3$. If $j \in I_2 = \{i, j\}$, it is easy to check that $\mathbf{D}_{I_2}^T \mathbf{D}_{\cdot j} = \mathbf{1}^T \mathbf{D}_{\cdot i} = 0$ as $t = 3$. For $t \geq 4$, since high-order orthogonal strength implies the low order ones, we only need to consider the situation of $k = t - 1$. And we can similarly obtain the conclusion by discussing j in I_k or not. □

With these results of FFD, if we determine the subdata matrix $\mathbf{X} \in \{0, 1\}^{m\times d}$ by exactly matching it to some resolution-R design $\mathbf{D} \in \{-1, 1\}^{m\times d}$ with the rule $\mathbf{D}_{\cdot j} = 2\mathbf{X}_{\cdot j} - \mathbf{1}$. We can show that such $\mathbf{X}$ is the optimal solution of Eq. (8) with weights $\mathbf{W} = \mathbf{1}$.

THEOREM 4.3. *For any* $\mathbf{X} \in \{0,1\}^{m\times d}$ *matching the resolution-R design with* $R \geq 3$*, we have* $\mathcal{L}_k(\mathbf{W}, \mathbf{X}) = 0$ *for any* $1 \leq k \leq R-2$ *and* $\mathbf{W} = \mathbf{1}$.

PROOF. Let $\mathbf{D} \in \{-1,1\}^{m\times d}$ be the resolution-R design matched with $\mathbf{X}$. For any $I_k \subseteq [d]\backslash\{j\}$ with a given j and feasible k, let $I_k = \{j_1, \ldots, j_k\}$ and we have

$$\mathbf{X}_{I_k} = \frac{1}{2^k}(\mathbf{D}_{\cdot j_1} + \mathbf{1}) \odot \cdots \odot (\mathbf{D}_{\cdot j_k} + \mathbf{1}) = \tfrac{1}{2^k}\left(\mathbf{1} + \textstyle\sum_{h=1}^{k}\sum_{\tilde{I}_h \subseteq I_k} \mathbf{D}_{\tilde{I}_h}\right), \tag{14}$$

where $\tilde{I}_h$ is the subset of I_k with cardinality h. When $\mathbf{W} = \mathbf{1}$, we have $\mathbf{W} \odot \mathbf{X}_{\cdot,j} = \mathbf{X}_{\cdot,j}$, $\mathbf{W} \odot (\mathbf{1} - \mathbf{X}_{\cdot,j}) = \mathbf{1} - \mathbf{X}_{\cdot,j}$ as well as $\mathbf{W}^T \cdot \mathbf{X}_{\cdot,j} = \mathbf{W}^T \cdot (\mathbf{1} - \mathbf{X}_{\cdot,j}) = m/2$ for $j = 1, \ldots, d$. The squared term in Eq. (8) becomes,

$$\begin{aligned}
&\left[\tfrac{\mathbf{X}_{I_k}^T \cdot (\mathbf{W}\odot\mathbf{X}_{\cdot,j})}{\mathbf{W}^T\cdot\mathbf{X}_{\cdot,j}} - \tfrac{\mathbf{X}_{I_k}^T \cdot (\mathbf{W}\odot(\mathbf{1}-\mathbf{X}_{\cdot,j}))}{\mathbf{W}^T\cdot(\mathbf{1}-\mathbf{X}_{\cdot,j})}\right]^2 \\
=&\left[\tfrac{2}{m}\mathbf{X}_{I_k}^T\,(2\mathbf{X}_{\cdot j} - \mathbf{1})\right]^2 = \left(\tfrac{2}{m}\mathbf{X}_{I_k}^T\mathbf{D}_{\cdot j}\right)^2 \\
=&\tfrac{1}{4^{k-1}m^2}\left(\mathbf{1}^T\mathbf{D}_{\cdot j} + \textstyle\sum_{h=1}^{k}\sum_{\tilde{I}_h\subseteq I_k}\mathbf{D}_{\tilde{I}_h}^T\mathbf{D}_{\cdot j}\right)^2 = 0,
\end{aligned} \tag{15}$$

where the last equality follows Theorem 2. Specifically, when $k = 1$ or $R = 3$, we have $\tilde{I}_h = I_k = \{j_k\}$ with $j_k \neq j$ and the last equality becomes zero according to the case a) in Theorem 2. Similar results can be derived for $2 \leq k \leq R-2$ ($R \geq 4$) following the case b). Consequently, it is evident that $\mathcal{L}_k(\mathbf{1}, \mathbf{X})$ equals to zero for any $1 \leq k \leq R-2$ ($R \geq 3$). □

This theorem establishes the key connection between FFD and the global balancing loss in the stable prediction problem. Note that the theorem also holds for $\mathbf{W} = \alpha\mathbf{1}, \forall\alpha > 0$. The theorem reveals that higher resolution design can lead to more stable outcomes, as the lower-order confounding effects are removed by the perfect balance. Since a higher resolution design would require a larger run size m. In the present paper, we use resolution-5 design as a subsampling template, which ensures $\mathcal{L}_k(\mathbf{1}, \mathbf{X}) = 0$ for $k = 1, 2, 3$. The template with a given m can be easily generated from open source packages, such as `FrF2` in R [18]. As for m, it should be the power of 2 and no less than 2^{R-1}. One may determine it based on either available templates or practical demands. We choose $m = 128$ in this work as it is empirically sufficient to achieve the stable prediction for all numerical studies. It turns out the computational cost based on subsampled data is highly reduced, as m is typically much smaller than the full data size n.

5 BALANCE-SUBSAMPLED STABLE PREDICTION ALGORITHM

BSSP algorithm consists of an FFD-based subsampling method and a subsampled learning model. We first introduce the specific subsampling algorithm that is feasible for general situations. To obtain a balanced subdata with deconfounded variables, we propose a matching algorithm based on the FFD template. Given a resolution-R design $\mathbf{D} \in \{-1,1\}^{m\times d}$, we transfer its encoding into $\{0,1\}$. Then we select the samples from $\mathbf{M} = \{\mathbf{X} \in \{0,1\}^{n\times d}, Y \in \mathbb{R}^n\}$ if some row in $\mathbf{D}$ can match the one in $\mathbf{X}$. The matching process is described in Algorithm 1.

However, it may not be easy to achieve a perfect matching and thus non-confounding properties in practice. Note that Lemma 4.1 implies the orthogonality is invariant to the column permutation of $\mathbf{D}$, which we may denote as $\mathscr{D}$ with the cardinality $d!$. All the designs in $\mathscr{D}$ share the same orthogonal properties as the template design $\mathbf{D}$. Hence, we may find

Algorithm 1 Sample_Matching Algorithm

Input: Observed samples $\mathbf{M} = \{\mathbf{X} \in \{0,1\}^{n\times d}, Y \in \mathbb{R}^n\}$ and a design $\mathbf{D} \in \{0,1\}^{m\times d}$
Output: A subset of samples $\mathbf{M}_{\text{sub}}$

```
Set M_sub = ∅
for each row/sample D_{i,·} ∈ D do
    if D_{i,·} == X_{j,·} then
        M_sub = M_sub ∪ (X_{j,·}, Y_j)
        break
    end if
end for
return M_sub
```

a better design $\mathbf{D}'$ in $\mathscr{D}$ such that all its design points can be matched to the observed samples.

When none of the designs in $\mathscr{D}$ can fully match the observed samples in $\mathbf{X}$, we propose the following criterion for measuring the degree of confounding byr $\mathbf{M}_{\text{sub}} = \{\widetilde{\mathbf{X}}, \widetilde{Y}\}$:

$$\psi(\mathbf{M}_{\text{sub}}) = \sum_{j=1}^{R-1} \rho^j A_j(\widetilde{\mathbf{X}}), \quad 0 < \rho < 1, \tag{16}$$

where $A_j(\widetilde{\mathbf{X}})$ refers to the generalized wordlength in Eq. (10) and ρ is a parameter for exponentially weighing. In the ideal case, we have $\psi(\mathbf{M}_{\text{sub}}) = 0$ according to the definition of resolution-R design. Note that $A_j(\widetilde{\mathbf{X}})$ reflects the severity of order-j confounding effects and is invariant to the design encoding. Motivated by the famous *effect hierarchy principle* [53]: (i) lower-order effects are more likely to be important; and (ii) effects with the same order are equally likely to be important, we set $\rho = 0.9$ to assign more weights to the lower-order effects. A simulation is performed shown by Fig. 2 to validate that $\psi(\mathbf{M}_{\text{sub}})$ can measure the deviation of $\widetilde{\mathbf{X}}$ to the FFD, where we calculate the $\psi(\mathbf{M}_{\text{sub}})$ on random subsets of a 128-run resolution-5 design with different sizes and 100 replications. As we can see, the measure goes to 0 without any variation when $\widetilde{\mathbf{X}}$ is close to the template design. Based on Eq. (16), one can calculate the confounding measure for each $\mathbf{D}$ in $\mathscr{D}$. Then, we can rank these subdata candidates and select the one with the minimal ψ-value. The details of the complete subsampling method are described in Algorithm 2.

With the balance-subsampled data $\mathbf{M}_{\text{sub}}$ from Algorithm 2, one can directly run a machine learning model for prediction, including regression for continuous outcome Y and classification for categorized outcome Y. In this work, we simply consider the typical linear regression and logistic model with the original linear features of $\widetilde{\mathbf{X}}$.

6 EXPERIMENTS

In this section, we evaluate the performance of our algorithm on both synthetic and real-world datasets.

6.1 Baseline Methods

We use the following three methods as the baselines for comparison.

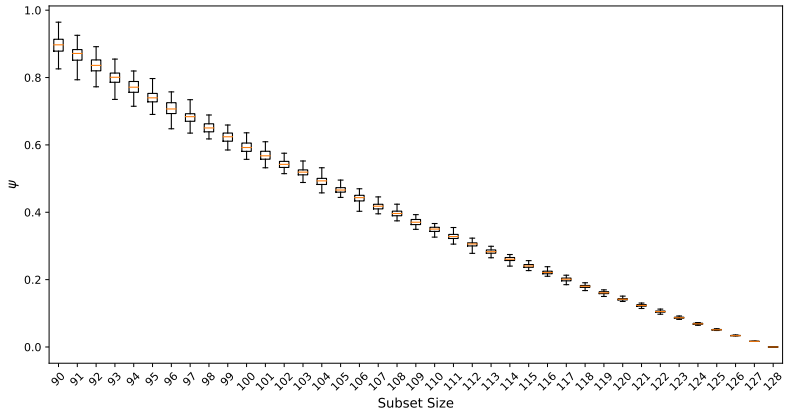


Fig. 2. $\psi(\mathbf{M}_{\text{sub}})$ on random subsets of a 128-run resolution-5 FFD with different subdata sizes.

Algorithm 2 FFD-Based Subsampling Algorithm

Input: Observed samples $\mathbf{M} = \{\mathbf{X} \in \{0,1\}^{n\times d}, Y \in \mathbb{R}^n\}$ and a resolution-R design $\mathbf{D} \in \{0,1\}^{m\times d}$.
Output: A subset of samples $\mathbf{M}_{\text{sub}}$

Set $\mathbf{M}_{\text{sub}} = \emptyset$
Generate a design set $\mathscr{D}$ by column permutation on $\mathbf{D}$,
for $\mathbf{D}' \in \mathscr{D}$ **do**
    $\mathbf{M}'_{\text{sub}} = Sample\_Matching\ (\mathbf{M}, \mathbf{D}')$
    Calculate its confounding measure $\psi(\mathbf{M}'_{\text{sub}})$
    if $\psi(\mathbf{M}'_{\text{sub}}) < \psi(\mathbf{M}_{\text{sub}})$ **then**
        Let $\mathbf{M}_{\text{sub}} = \mathbf{M}'_{\text{sub}}$
    end if
    if $\psi(\mathbf{M}_{\text{sub}}) == 0$ **then** ▷ all samples in $\mathbf{D}'$ are matched
        break
    end if
end for
return $\mathbf{M}_{\text{sub}}$

- Logistic Regression (LR): A baseline for the classification task. It needs to assume there is no distribution bias between training and test data, and therefore cannot address the stable prediction problem.
- Ordinary Least Square (OLS): A baseline for regression task. It also needs to assume both training and test data have the same distribution, and therefore cannot address the stable prediction problem.
- Global Balancing Regression (GBR) [24]: It learns the global weights for all the samples in order to make the variables approximately mutually non-confounding. It then performs weighted classification and regression and may approximately address the stable prediction problem.

Here, we use LR for the classification task and OLS for regression task, while GBR and our BSSP algorithm can be applied to both classification and regression tasks. Specifically, the LASSO regularizer [48] is configured in all above methods.

6.2 Evaluation Metrics

In our experiments, we perform stable learning for both classification and regression tasks. To evaluate the performance, we use Root Mean Squared Error ($RMSE$), β_Error, $Average\_Error$, and $Stability\_Error$ as evaluation metrics. Their definitions are listed as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(Y_k - \hat{Y}_k)^2}, \tag{17}$$

where n is the sample size, $\hat{Y}_k$ and Y_k refer to the predicted and true outcomes for sample k;

$$\beta\_Error = \|\beta - \hat{\beta}\|_1, \tag{18}$$

where $\hat{\beta}$ and β represent the estimated and true regression coefficients. $Average\_Error$, and $Stability\_Error$ are defined in Eq. (1) and (2), where $Error(\mathbf{M}^e)$ is defined as $RMSE(\mathbf{M}^e)$.

6.3 Experiments on Synthetic Datasets

6.3.1 *Stable Prediction for Regression Task.*

Datasets. Firstly, we generate observed binary predictors $\mathbf{X} = \{\mathbf{S}_{\cdot,1}, \cdots, \mathbf{S}_{\cdot,d_s}, \mathbf{V}_{\cdot,1}, \cdots, \mathbf{V}_{\cdot,d_v}\}$ with independent Gaussian distribution:

$$\hat{\mathbf{S}}_{\cdot,1}, \cdots, \hat{\mathbf{S}}_{\cdot,d_s}, \hat{\mathbf{V}}_{\cdot,1}, \cdots, \hat{\mathbf{V}}_{\cdot,d_v} \overset{iid}{\sim} \mathcal{N}(0,1), \tag{19}$$

where $d_s + d_v = d$, and $\mathbf{S}_{\cdot,j}$ represents the j^{th} variable in $\mathbf{S}$. To make $\mathbf{X}$ binary, we let $\mathbf{X}_{\cdot,j} = 1$ if $\hat{\mathbf{X}}_{\cdot,j} \geq 0$, otherwise $\mathbf{X}_{\cdot,j} = 0$. Then, we generate continuous response variable Y by the following function:

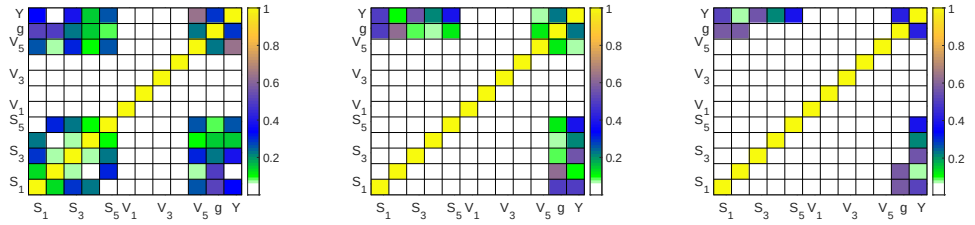
$$Y = f(\mathbf{S}) + \varepsilon = [\mathbf{S}, \mathbf{V}] \cdot [\beta_s, \beta_v]^T + \mathbf{S}_{\cdot,1}\mathbf{S}_{\cdot,2} + \varepsilon, \tag{20}$$

where $\beta_s = \{\frac{1}{3}, -\frac{2}{3}, 1, -\frac{1}{3}, \frac{2}{3}, -1, \cdots\}$, $\beta_v = \vec{0}$, and $\varepsilon = \mathcal{N}(0, 0.3)$. The term $\mathbf{S}_{\cdot,1}\mathbf{S}_{\cdot,2}$ refers to the omitted non-linear term $g(\cdot)$ in Eq. (4).

Various Environments. To test the stability of all algorithms, we need to generate a set of environments, each with a distinct joint distribution $P(\mathbf{X}, Y)$, while preserving Assumption 1 (and in particular, $P(Y|\mathbf{S})$). Under Theorem 3.1, we generate different environments in our experiments by varying $P(\mathbf{V}|\mathbf{S})$. For simplicity, we only change $P(\mathbf{V}_b|\mathbf{S})$ on a subset of noisy features $\mathbf{V}_b \subseteq \mathbf{V}$, where the dimension of $\mathbf{V}_b$ is $0.1 * d$.

Specifically, we vary $P(\mathbf{V}_b|\mathbf{S})$ via biased sample selection with a bias rate of $r \in [-3, -1) \cup (1, 3]$. For each sample, we select it with probability $Pr = \prod_{\mathbf{V}_i \in \mathbf{V}_b} |r|^{-5*D_i}$, where $D_i = |f(\mathbf{S}) - sign(r) * \mathbf{V}_i|$ with $f(\mathbf{S})$ defined in Eq. (20). If $r > 0$, $sign(r) = 1$; otherwise, $sign(r) = -1$.

Note that $r > 1$ corresponds to positive unstable correlation between Y and $\mathbf{V}_b$, while $r < -1$ corresponds to negative unstable correlation between Y and $\mathbf{V}_b$. The larger value of $|r|$, the stronger correlation between $\mathbf{V}_b$ and Y. Different values of r refer to different environments, hence we can generate different environments by varying $P(\mathbf{V}_b|\mathbf{S})$.



(a) Raw data (b) Weighted by GBR (c) Subsampled by BSSP

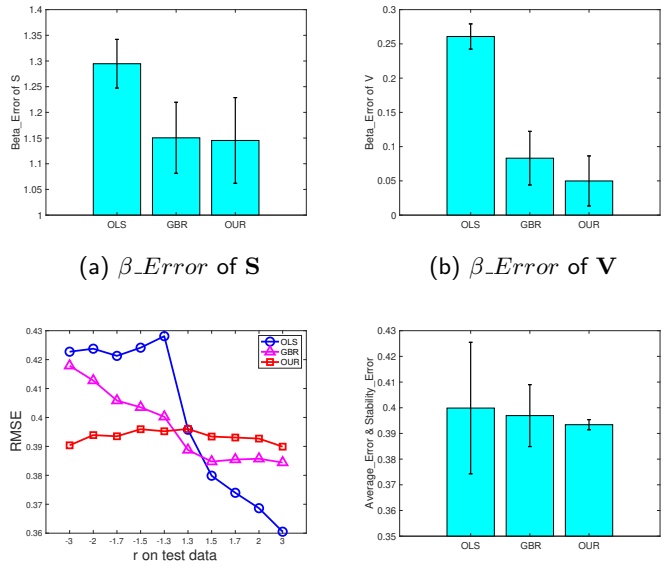
Fig. 3. Pearson correlation coefficients among variables: a) on raw data; b) on weighted data; c) on subsampled data.

Results. In experiments, we evaluate the performance of all algorithms from two aspects, including accuracy on parameter estimation and stability on prediction across unknown test data. To measure the accuracy of parameter estimation, we train all models on one training dataset with a specific bias rate r_{train}. We carry out model training for 50 times independently with different training data from the same bias rate r_{train}, and report the mean and variance of β_Error on stable features $\mathbf{S}$ and noisy features $\mathbf{V}$. To evaluate the stability of prediction, we test all models on various test environments with different bias rate $r_{test} \in [-3, -1) \cup (1, 3]$. For each r_{test}, we generate 50 different test datasets and report the mean of RMSE. With RMSE from all test environments, we report $Average\_Error$ and $Stability\_Error$ to evaluate the stability of prediction across unknown test environments.

Before reporting the experimental results, we compare the Pearson correlation coefficients between any two variables on a) raw data, b) the weighted data by global balancing method [24] and c) the subdata by our algorithm in Fig. 3. From the result, we can find that in the raw data (Fig. 3a), the noisy feature $\mathbf{V}_5$ is correlated with some stable features $\mathbf{S}$, and highly correlated with both omitted nonlinear term g and outcome Y. Hence, the estimated coefficient of $\mathbf{V}_5$ in traditional regression models, such as LR and OLS, would be large, which should be *zero* in a correctly specified model, leading to unstable prediction. In the weighted data by global balancing method (Fig. 3b), the sample weights learned from global balancing can clearly remove the correlation among predictors $\mathbf{X}$, especially the correlation between noisy feature $\mathbf{V}_5$ and stable features $\mathbf{S}$. But we can find that noisy feature $\mathbf{V}_5$ is still correlated with both omitted nonlinear term g and outcome Y, leading to imprecise estimation on coefficients of noisy features and resulting in unstable prediction. The main reason is that the global balancing method is focused on first order confounding while ignoring higher-order confounding between $\mathbf{V}_5$ and g. In the subsampled data by our algorithm (Fig. 3c), we can find that not only the correlation among predictors $\mathbf{X}$, but also the correlation between noisy feature $\mathbf{V}_5$ and the omitted nonlinear term g and outcome Y are clearly removed. Hence, our algorithm can estimate the coefficient of both $\mathbf{S}$ and $\mathbf{V}$ more precisely. This is the key reason that our algorithm can make more stable predictions across unknown test environments.

We report the results of parameter estimation and stable prediction in Fig. 4, for which the sample size is $n = 2000$ and the dimensionality of variables is $d = 10$. From the results, we have the following observations and analysis:

- OLS cannot address the stable prediction problem. The reason is that OLS is biased on both β_S and β_V estimation since the confounding or spurious correlation between



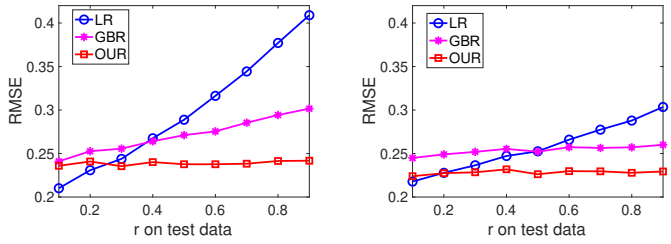
(a) β_Error of $\mathbf{S}$ (b) β_Error of $\mathbf{V}$

(c) $RMSE$ over r (d) $Average\_Error$ (green bar) & $Stability\_Error$ (black line)

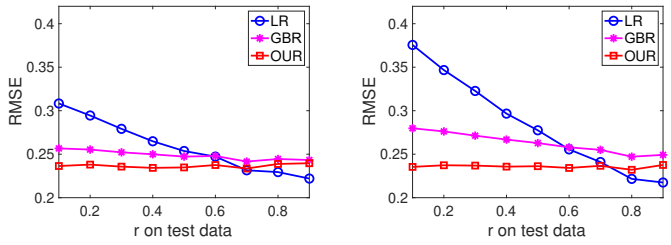
Fig. 4. Results of regression. All the models are trained with $n = 2000$, $d = 10$ and $r_{train} = 2.0$. The parameter $r \in [-3, -1) \cup (1, 3]$ refers to the rate of bias selection when generating a regression dataset. A large absolute value of r corresponds to stronger confounding effects. The training and test datasets with the same value of r are generated from the same environment.

$\mathbf{S}$ and $\mathbf{V}$. Moreover, OLS will often predict large effects of the unstable features, which leads to instability across environments.

- With considering variable deconfounding loss by sample reweighting, GBR method achieves better performance than OLS in the parameter estimation on both stable features and unstable features (See Fig. 4a & 4b). Hence, GBR method can obtain a more stable prediction than OLS (The $Average\_Error$ and $Stability\_Error$ of GBR method in Fig. 4d is clearly smaller than OLS method).
- The performance of our algorithm is worse than baseline when $r_{test} \geq 1.3$ on test data in Fig. 4c, but much better than baselines when $r_{test} \leq -1.3$. This is because the spurious correlations between $\mathbf{V}_b$ and Y are similar between training data ($r_{train} = 2.0$) and test data when $r_{test} \geq 1.3$, and that correlation can be exploited in prediction; in this setting, $\mathbf{V}$ is useful to proxy for omitted functions of $\mathbf{S}$. However, when $r_{test} \leq -1.3$, using $\mathbf{V}$ for prediction induces obvious instability.
- We also provide a mathematical perspective to analyze the results shown in Fig. 4c. Specifically, concerning Eq. (4), we have the OLS estimator for β_S and β_V as



(a) $r_{train} = 0.15$ (b) $r_{train} = 0.25$



(c) $r_{train} = 0.75$ (d) $r_{train} = 0.85$

Fig. 5. Results of classification on various test datasets by varying r_{train}. The parameter $r \in (0, 1)$ corresponds to the rate of bias selection when generating a classification dataset. Generally, a large value of $|r - 0.5|$ corresponds to stronger confounding effects. The training and test datasets with the same selection rate $r_{train} = r_{test}$ are generated from the same environment.

follows,

$$\hat{\beta}_{V_{OLS}} = \beta_V + \left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{V}_i^T\mathbf{V}_i\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{V}_i^T g\left(\mathbf{S}_i\right)\right) + \left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{V}_\mathbf{i}^T\mathbf{V}_i\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{V}_i^T\mathbf{S}_i\right)\left(\beta_S - \hat{\beta}_{S_{OLS}}\right), \tag{21}$$

$$\hat{\beta}_{S_{OLS}} = \beta_S + \left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{S}_i^T\mathbf{S}_i\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{S}_i^T g\left(\mathbf{S}_i\right)\right) + \left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{S}_i^T\mathbf{S}_i\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{S}_i^T\mathbf{V}_i\right)\left(\beta_V - \hat{\beta}_{V_{OLS}}\right), \tag{22}$$

where n denotes the sample size. It is easy to see that OLS is very sensitive to the confounding effects among $\mathbf{V}$, $\mathbf{S}$ and $g(\mathbf{S})$ under different environments in terms of $\sum_{i=1}^{n}\mathbf{V}_i^T g\left(\mathbf{S}_i\right)/n$ and $\sum_{i=1}^{n}\mathbf{S}_i^T\mathbf{V}_i/n$. In contrast, GBR can alleviate the confounding effects between $\mathbf{V}$ and $\mathbf{S}$ via reweighting observations, which leads to a more stable performance. Whereas our proposed approach additionally reduces the confounding effects between $\mathbf{V}$ and $g(\mathbf{S})$ relative to GBR through subsampling, and hence should achieve the most stable predictions across various environments.

- Comparing with baselines, our algorithm achieves the best stable prediction across unknown test environments. Smart sampling in our algorithm can ensure the orthogonality and nonconfounding properties among variables. Moreover, the noisy features will become exactly uncorrelated with the omitted non-linear term and outcome variable. Therefore, our algorithm avoids using noisy features to proxy for omitted nonlinear functions of the stable features, ensuring less bias in the estimation of the effect of both stable features and noisy features, and improving the stability of prediction.

6.3.2 *Stable Prediction for Classification Task.*

Datasets. Firstly, we generate observed binary predictors $\mathbf{X} = \{\mathbf{S}_{\cdot,1}, \cdots, \mathbf{S}_{\cdot,d_s}, \mathbf{V}_{\cdot,1}, \cdots, \mathbf{V}_{\cdot,d_v}\}$ with independent Gaussian distributions as:

$$\hat{\mathbf{S}}_{\cdot,1}, \cdots, \hat{\mathbf{S}}_{\cdot,d_s}, \hat{\mathbf{V}}_{\cdot,1}, \cdots, \hat{\mathbf{V}}_{\cdot,d_v} \overset{iid}{\sim} \mathcal{N}(0,1), \tag{23}$$

where $d_s + d_v = d$, and $\mathbf{S}_{\cdot,j}$ represents the j^{th} variable in $\mathbf{S}$. To make $\mathbf{X}$ binary, we let $\mathbf{X}_{\cdot,j} = 1$ if $\hat{\mathbf{X}}_{\cdot,j} \geq 0$, otherwise $\mathbf{X}_{\cdot,j} = 0$.

Then, we generate binary response variable Y with the function as following:

$$Y = 1/(1 + \exp(-\textstyle\sum_{\mathbf{X}_{\cdot,i} \in \mathbf{S}_l} \alpha_i \cdot \mathbf{X}_{\cdot,i} - \sum_{\mathbf{X}_{\cdot,j} \in \mathbf{S}_n} \beta_j \cdot \mathbf{X}_{\cdot,j} \cdot \mathbf{X}_{\cdot,j+1})) + \mathcal{N}(0, 0.2), \tag{24}$$

where we separate the stable features $\mathbf{S}$ into two parts, linear part $\mathbf{S}_l$ and non-linear part $\mathbf{S}_n$. $\alpha_i = (-1)^i \cdot (mod(i,3)+1) \cdot d/3$ and $\beta_j = d/2$, where function $mod(x,y)$ returns the modulus after division of x by y. To make Y binary, we set $Y = 1$ when $Y \geq 0.5$, otherwise $Y = 0$.

Various Environments. We generate various environments by varying $P(Y|\mathbf{V})$ via biased sample selection with a bias rate $r \in (0,1)$. Specifically, we select a sample with probability r if its noisy features equal to the response variable, that is $\mathbf{V} = Y$; otherwise, we select it with probability $1 - r$, where $r > 0.5 (< 0.5)$ corresponds to a positive (negative) correlation between Y and $\mathbf{V}$.

Results. In our experiments, we fix the sample size $n = 2000$ and dimensions of variables $d = 10$, but generate different synthetic data by varying bias rate on training data $r_{train} = \{.15, .25, .75, .85\}$. We report the results in Fig. 5. From the results, we have the following observations and analysis:

- The methods LR can not address the stable prediction problem in all settings, since it cannot remove the confounding or spurious correlation between $\mathbf{V}$ and Y during model training, and often predict large effects of the noisy features $\mathbf{V}$, leading to instability across test environments.
- With making variables become approximately non-confounding by sample weighting, GBR method obtains a more stable prediction than LR method, especially when the bias rate on training r_{train} is closer to 0.5. But as increasing of $|r_{train} - 0.5|$, its performance on stable prediction becomes worse. The main reason is that r_{train} is closer to 0.5 refers to the spurious correlation between $\mathbf{V}$ and Y is not such strong, and the global balancing regularizer can remove those correlations and ensure accurate parameter estimation. But as spurious correlation become stronger, GBR cannot fully remove them.
- Comparing with baselines, our method achieves the most stable prediction in different settings. Smart sampling based on FFD ensures the non-confounding among variables,

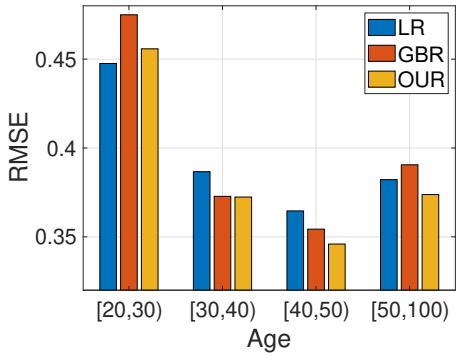


Fig. 6. Prediction across environments separated by age. The models are trained on dataset where uses' $Age \in [20, 30)$, but tested on various datasets with different users' age range.

hence regression on those samples helps to accurately estimate the effect of both stable features and noisy features for stable prediction.

- By varying the bias rate on training r_{train}, the RMSE of our algorithm is consistently stable and small across environments. Moreover, our algorithm makes greater improvements when r is farther from 0.5, i.e., stronger of the spurious correlation between $\mathbf{V}$ and Y.

6.4 Experiments on Real Datasets

We apply the proposed BSSP algorithm on two real-world datasets, including WeChat advertising dataset (classification) and Parkinson's telemonitoring data (regression).

6.4.1 Wechat Advertising Dataset.

Dataset. To check the performance of our algorithm in the classification setting, we apply it to a real online advertising dataset, which is collected from Tencent WeChat App[4] during September 2015 and used in [24] for stable prediction. In WeChat, each user can share (receive) posts to (from) his/her friends as like the Twitter and Facebook. Then the advertisers could push their advertisements to users, by merging them into the list of the user's wall posts. For each advertisement, there are two types of feedbacks: "Like" and "Dislike". When the user clicks the "Like" button, his/her friends will receive the advertisements with this action.

The Wechat advertising campaign used in our paper is about the LONGCHAMP handbags for young women.[5] This campaign contains 14,891 user feedbacks with Like and 93,108 Dislikes. For each user, we have their features including (1) demographic attributes, such as age, gender, (2) number of friends, (3) device (iOS or Android), and (4) the user settings on WeChat, for example, whether allowing strangers to see his/her album and whether installing the online payment service.

Experimental Settings. In our experiments, we set $Y_i = 1$ if user i likes the ad, otherwise $Y_i = 0$. For non-binary features, we dichotomize them around their mean value. Specifically,

[4]http://www.wechat.com/en/
[5]http://en.longchamp.com/en/womens-bags

we only preserve users' features which satisfied $0.2 \leq \frac{\#\{x=1\}}{\#\{x=1\}+\#\{x=0\}} \leq 0.8$. Finally, our dataset contains 10 user features as predictor variables and user feedback as the outcome variable, all of them are binary.

To test the stability of all methods, we generate different environments by dataset separation with users' feature. Specifically, we separate the whole dataset into 4 parts by users' age, including $Age \in [20, 30)$, $Age \in [30, 40)$, $Age \in [40, 50)$ and $Age \in [50, 100)$. In our experiments, we trained all models with data from environment $Age \in [20, 30)$ but tested them on all 4 environments.

Results. We plot the results in Figure 6. From the results, we can obtain that our proposed algorithm achieves comparable results to the baseline OLS on test environment with $Age \in [20, 30)$, where the variables' distributions are similar or even the same with the one on the training environment. On the other three test environments, whose distributions differ from the training environment, our algorithm achieves the best prediction performance. Another important observation is that the performance of our algorithm is always better than the global balancing method. The main reason is that the global balancing method cannot address the high-order confounding among variables, while our smart sampling and stable prediction algorithm can as guaranteed with theorem 4.3.

6.4.2 *Parkinson's Telemonitoring Dataset.*

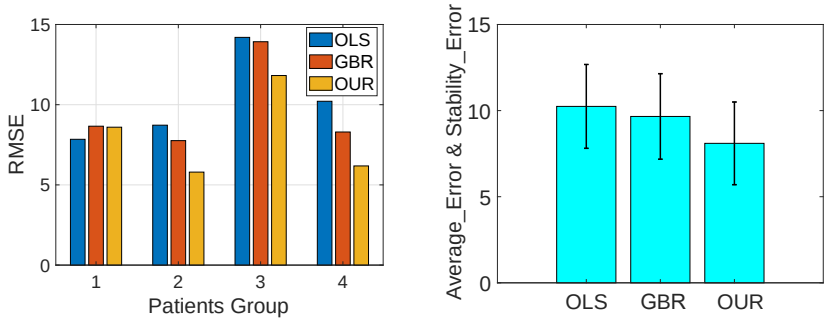
Dataset. To test our algorithm in a regression setting, we apply it to a Parkinson's telemonitoring dataset[6], which has been wildly used for domain generalization [6, 38] task and other regression task [49]. The dataset is composed of biomedical voice measurements from 42 patients with early-stage Parkinson's disease recruited for a six-month trial of a telemonitoring device for remote symptom progression monitoring. For each patient, there are around 200 recordings, which were automatically captured in the patients' homes. The aim is to predict the clinician's motor and total UPDRS scoring of Parkinson's disease symptoms from patients' features, including their age, gender, test time and many other measures.

Experimental Settings. In our experiments, we set the outcome variables Y as motor UPDRS scoring and total UPDRS scoring separately. For those non-binary features, we dichotomize them around their mean value. Finally, we selected 10 patients features as predictors $\mathbf{X}$, including age, gender, test time, Jitter:PPQ5 (a measure of variation in fundamental frequency), Shimmer:APQ5 (a measure of variation in amplitude), RPDE (a nonlinear dynamical complexity measure), DFA (signal fractal scaling exponent), PPE (a nonlinear measure of fundamental frequency variation), NHR and HNR (two measures of ratio of noise to tonal components in the voice).

To test the stability of all algorithms, we generate various environments by data separation on different patients. Specifically, we separate the whole 42 patients into 4 patients' groups, including group 1 with recordings from 21 patients, and the other three groups (group 2, 3 and 4) are all with recordings from different 7 patients. In our experiments, we trained all models with data from environment of group 1, but tested them on all 4 patients' groups.

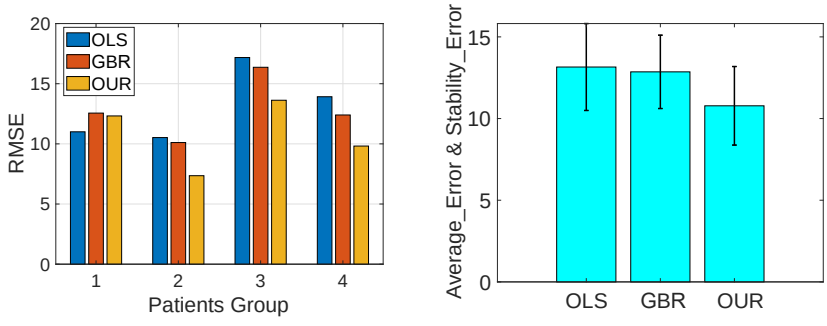
Results. We report the experimental results in Figure 7 when we set outcome as patients' motor UPDRS score, and Figure 8 for patients' total UPDRS score prediction. From Figures 7a & 8a, we can observe that the predictive results of our algorithm and GBR method are

[6] https://archive.ics.uci.edu/ml/datasets/parkinsons+telemonitoring



(a) $RMSE$ across patients' groups (b) $Average\_Error$ (green bar) & $Stability\_Error$ (black line)

Fig. 7. Prediction across patients' groups where outcome is motor UPDRS score. Models are trained on datasets from patients' group 1, but tested on datasets across patients' groups.



(a) $RMSE$ across patients' groups (b) $Average\_Error$ (green bar) & $Stability\_Error$ (black line)

Fig. 8. Prediction across patients' groups where outcome is total UPDRS score. Models are trained on datasets from from patients' group 1, but tested on datasets across patients' groups.

worse than OLS methods when the testing data are from patients' group 1. This is because the training and testing data are from the same environment, ensuring they have similar even the same covariates' distribution. But the performances of our algorithm and GBR method are much better than OLS on the other three environments, whose distribution might be different from the training environment. The results demonstrate that non-confounding features by the global balancing or subsampling can help to address the agnostic distribution bias between training and testing environments, hence making better predictions across unknown testing environments. By comparing our algorithm with GBR, we find that the performance of our algorithm is always better than global balancing method. The main reason is that our algorithm can ensure higher-order deconfounding among variables if we can make exactly subsampling based on FFD, while the GBR only focuses on the first confounding among variables.

To explicitly demonstrate the advantage of our proposed algorithm, we report *Average_Error* and *Stability_Error* in Figure 7b for motor UPDRS score prediction and Figure 8b for total UPDRS score prediction. The results show that our algorithm makes the most stable prediction (with smallest *Average_Error* and smallest *Stability_Error*) across all environments.

7 CONCLUSION

This paper addresses the problem of stable prediction across unknown environments. We propose a subsampling method to reduce the spurious correlation between the noisy features and the outcome variable. The subsampling method uses fractional factorial design as a matching template, which promotes the non-confounding properties among features. We develop a new confounding measure for subsample selection. The proposed BSSP method can be used in both regression and classification tasks. Extensive experiments on synthetic and real-world datasets have clearly demonstrated the advantages of the proposed method for stable prediction. Our future work will extend the BSSP method from binary features to multi-category and continuous features, for which multi-level fractional factorial designs and uniform space-filling designs can be considered as the matching templates. We will also explore how the balance-subsampling idea can be used for stable prediction with the neural network models and the tensor data.

ACKNOWLEDGMENTS

This work was supported in part by Key Research and Development Projects of the Ministry of Science and Technology (No. 2020YFC0832500), National Natural Science Foundation of China (No. 61625107, No. 62006207), National Key Research and Development Program of China (No. 2018AAA0101900), the Fundamental Research Funds for the Central Universities and Zhejiang Province Natural Science Foundation (No. LQ21F020020)

REFERENCES

[1] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[2] S. Athey, G. W. Imbens, and S. Wager. Approximate residual balancing: debiased inference of average treatment effects in high dimensions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(4):597–623, 2018.

[3] P. C. Austin. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate Behavioral Research*, 46(3):399–424, 2011.

[4] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010.

[5] S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10(Sep):2137–2155, 2009.

[6] G. Blanchard, A. A. Deshmukh, U. Dogan, G. Lee, and C. Scott. Domain generalization by marginal transfer learning. *arXiv preprint arXiv:1711.07910*, 2017.

[7] J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, 2006.

[8] G. E. Box, J. S. Hunter, and W. G. Hunter. Statistics for experimenters. Wiley Hoboken, NJ, USA, 2005.

[9] R. Chattopadhyay, Q. Sun, W. Fan, I. Davidson, S. Panchanathan, and J. Ye. Multisource domain adaptation and its application to early detection of fatigue. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(4):1–26, 2012.

[10] I. Chaturvedi, E. Cambria, S. Cavallari, and R. E. Welsch. Genetic programming for domain adaptation in product reviews. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8. IEEE,

2020.
[11] A. Dey and R. Mukerjee. *Fractional Factorial Plans*, volume 496. John Wiley & Sons, 2009.
[12] P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlós. Faster least squares approximation. *Numerische Mathematik*, 117(2):219–249, 2011.
[13] J. Duchi and H. Namkoong. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018.
[14] C. Fong, C. Hazlett, K. Imai, et al. Covariate balancing propensity score for a continuous treatment: application to the efficacy of political advertisements. *The Annals of Applied Statistics*, 12(1):156–177, 2018.
[15] A. Fries and W. G. Hunter. Minimum aberration 2^{k-p} designs. *Technometrics*, 22(4):601–608, 1980.
[16] W. Fu, B. Xue, X. Gao, and M. Zhang. Genetic programming based transfer learning for document classification with self-taught and ensemble learning. In *2019 IEEE Congress on Evolutionary Computation*, pages 2260–2267, 2019.
[17] W. Fu, B. Xue, M. Zhang, and X. Gao. Transductive transfer learning in genetic programming for document classification. In *Asia-Pacific Conference on Simulated Evolution and Learning*, pages 556–568. Springer, 2017.
[18] U. Grönmping. R package frf2 for creating and analyzing fractional factorial 2-level designs. *Journal of Statistical Software*, 56(1):1–56, 2014.
[19] J. Hainmueller. Entropy balancing for causal effects: A multivariate reweighting method to produce balanced samples in observational studies. *Political Analysis*, 20(1):25–46, 2012.
[20] A. S. Hedayat, N. J. A. Sloane, and J. Stufken. *Orthogonal Arrays: Theory and Applications*. Springer Science & Business Media, 2012.
[21] L. Hou, X.-y. Luo, Z.-y. Wang, and J. Liang. Representation learning via a semi-supervised stacked distance autoencoder for image classification. *Frontiers of Information Technology & Electronic Engineering*, 21(7):1005–1018, 2020.
[22] G.-B. Huang, E. Cambria, K.-A. Toh, B. Widrow, and Z. Xu. New trends of learning in computational intelligence [guest editorial]. *IEEE Computational Intelligence Magazine*, 10(2):16–17, 2015.
[23] J. Jiang and C. Zhai. Instance weighting for domain adaptation in nlp. In *45th Annual Meeting of the Association for Computational Linguistics, ACL 2007*, pages 264–271, 2007.
[24] K. Kuang, P. Cui, S. Athey, R. Xiong, and B. Li. Stable prediction across unknown environments. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1617–1626. ACM, 2018.
[25] K. Kuang, P. Cui, B. Li, M. Jiang, Y. Wang, F. Wu, and S. Yang. Treatment effect estimation via differentiated confounder balancing and regression. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(1):1–25, 2019.
[26] K. Kuang, P. Cui, B. Li, M. Jiang, and S. Yang. Estimating treatment effect in the wild via differentiated confounder balancing. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 265–274. ACM, 2017.
[27] K. Kuang, P. Cui, B. Li, M. Jiang, S. Yang, and F. Wang. Treatment effect estimation with data-driven variable decomposition. In *AAAI*, pages 140–146, 2017.
[28] K. Kuang, L. Li, Z. Geng, L. Xu, K. Zhang, B. Liao, H. Huang, P. Ding, W. Miao, and Z. Jiang. Causal inference. *Engineering*, 6(3):253–263, 2020.
[29] J. Liu, Z. Hu, P. Cui, B. Li, and Z. Shen. Heterogeneous risk minimization. *arXiv preprint arXiv:2105.03818*, 2021.
[30] J. Liu, Z. Shen, P. Cui, L. Zhou, K. Kuang, and B. Li. Distributionally robust learning with stable adversarial training. *arXiv preprint arXiv:2106.15791*, 2021.
[31] M. Long, J. Wang, G. Ding, S. J. Pan, and S. Y. Philip. Adaptation regularization: A general framework for transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 26(5):1076–1089, 2013.
[32] M. López, A. Valdivia, E. Martínez-Cámara, M. V. Luzón, and F. Herrera. E2sam: evolutionary ensemble of sentiment analysis methods for domain adaptation. *Information Sciences*, 480:273–286, 2019.
[33] C. Lu and S. Wang. The general-purpose intelligent agent. *Engineering*, 6(3):221–226, 2020.
[34] C.-X. Ma and K.-T. Fang. A note on generalized aberration in factorial designs. *Metrika*, 53(1):85–93, 2001.
[35] P. Ma, M. W. Mahoney, and B. Yu. A statistical perspective on algorithmic leveraging. *The Journal of Machine Learning Research*, 16(1):861–911, 2015.

[36] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain. Sentic lstm: a hybrid network for targeted aspect-based sentiment analysis. *Cognitive Computation*, 10(4):639–650, 2018.
[37] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes*, volume 16. Elsevier, 1977.
[38] K. Muandet, D. Balduzzi, and B. Schölkopf. Domain generalization via invariant feature representation. In *Proceedings of the 30th International Conference on Machine Learning*, pages 10–18, 2013.
[39] S. J. Pan, Q. Yang, et al. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.
[40] J. Peters, P. Bühlmann, and N. Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012, 2016.
[41] K. Ren, T. Zheng, Z. Qin, and X. Liu. Adversarial attacks and defenses in deep learning. *Engineering*, 6(3):346–360, 2020.
[42] M. Rojas-Carulla, B. Schölkopf, R. Turner, and J. Peters. Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342, 2018.
[43] G. Rong, A. Mendez, E. B. Assi, B. Zhao, and M. Sawan. Artificial intelligence in healthcare: review and prediction case studies. *Engineering*, 6(3):291–301, 2020.
[44] P. R. Rosenbaum and D. B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.
[45] Z. Shen, P. Cui, K. Kuang, B. Li, and P. Chen. Causally regularized learning with agnostic data selection bias. In *Proceedings of the 26th ACM International Conference on Multimedia*, pages 411–419, 2018.
[46] S. K. Thompson. *Sampling*. Wiley Series in Probability and Statistics. Wiley, 3 edition, 2012.
[47] Q. Tian, K. Kuang, K. Jiang, F. Wu, and Y. Wang. Analysis and applications of class-wise robustness in adversarial training. 2021.
[48] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.
[49] A. Tsanas, M. A. Little, P. E. McSharry, and L. O. Ramig. Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests. *IEEE Transactions on Biomedical Engineering*, 57(4):884–893, 2009.
[50] H. Wang. More efficient estimation for logistic regression with optimal subsamples. *The Journal of Machine Learning Research*, 20(132):1–59, 2019.
[51] H. Wang, M. Yang, and J. Stufken. Information-based optimal subdata selection for big data linear regression. *Journal of the American Statistical Association*, 114(525):393–405, 2019.
[52] X. Wang, S. Fan, K. Kuang, C. Shi, J. Liu, and B. Wang. Decorrelated clustering with data selection bias. *arXiv preprint arXiv:2006.15874*, 2020.
[53] C. J. Wu and M. S. Hamada. *Experiments: Planning, Analysis, and Optimization*, volume 552. John Wiley & Sons, 2011.
[54] H. Xu and C. J. Wu. Generalized minimum aberration for asymmetrical fractional factorial designs. *The Annals of Statistics*, 29(4):1066–1077, 2001.
[55] A. Zhang, K.-T. Fang, R. Li, A. Sudjianto, et al. Majorization framework for balanced lattice designs. *The Annals of Statistics*, 33(6):2837–2853, 2005.
[56] A. Zhang, H. Zhang, and G. Yin. Adaptive iterative hessian sketch via a-optimal subsampling. *Statistics and Computing*, 30:1075–1090, 2020.
[57] M. Zhou, N. Duan, S. Liu, and H.-Y. Shum. Progress in neural nlp: modeling, learning, and reasoning. *Engineering*, 6(3):275–290, 2020.
[58] Y. Ziser and R. Reichart. Neural structural correspondence learning for domain adaptation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 400–410, 2017.
[59] Y. Ziser and R. Reichart. Deep pivot-based modeling for cross-language cross-domain transfer with minimal guidance. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 238–249, 2018.
[60] Y. Ziser and R. Reichart. Task refinement learning for improved accuracy and stability of unsupervised domain adaptation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5895–5906, 2019.
[61] J. R. Zubizarreta. Stable weights that balance covariates for estimation with incomplete outcome data. *Journal of the American Statistical Association*, 110(511):910–922, 2015.