# Generalization-Aware Structured Regression towards Balancing Bias and Variance

**Martin Pavlovski**[1,2], **Fang Zhou**[1], **Nino Arsov**[2], **Ljupco Kocarev**[2] and **Zoran Obradovic**[1]
[1] Center for Data Analytics and Biomedical Informatics, Temple University, Philadelphia, PA, USA
[2] Macedonian Academy of Sciences and Arts, Skopje, Republic of Macedonia
{martin.pavlovski, fang.zhou, zoran.obradovic}@temple.edu, {narsov, lkocarev}@manu.edu.mk

## Abstract

Attaining the proper balance between underfitting and overfitting is one of the central challenges in machine learning. It has been approached mostly by deriving bounds on generalization risks of learning algorithms. Such bounds are, however, rarely controllable. In this study, a novel bias-variance balancing objective function is introduced in order to improve generalization performance. By utilizing distance correlation, this objective function is able to indirectly control a stability-based upper bound on a model's expected true risk. In addition, the Generalization-Aware Collaborative Ensemble Regressor (GLACER) is developed, a model that bags a crowd of structured regression models, while allowing them to collaborate in a fashion that minimizes the proposed objective function. The experimental results on both synthetic and real-world data indicate that such an objective enhances the overall model's predictive performance. When compared against a broad range of both traditional and structured regression models GLACER was $\sim$10-56% and $\sim$49-99% more accurate for the task of predicting housing prices and hospital readmissions, respectively.

## 1 Introduction

One of the fundamental challenges in machine learning is to develop models that can learn from empirical evidence and make accurate predictions. Such a challenge requires a *design of learning algorithms capable of producing hypotheses (models) that 1) assimilate the empirical evidence, and 2) generalize well to unobserved data*. The former is controlled by the notion of *empirical risk* $R_{emp}$ (training error), while the latter depends on the *generalization risk* $R_{gen} = |R_{emp} - R_{true}|$ which determines whether $R_{emp}$ is a valid estimate of the *true unknown risk* $R_{true}$ (test error). A situation of high empirical risk $R_{emp}$ and low generalization risk $R_{gen}$ causes *underfitting*, characterized by the presence of high bias. On the other hand, the converse leads to *overfitting* indicating high variance. Therefore, in mathematical terms, the problem comes down to minimizing $R_{emp}$, while maintaining low $R_{gen}$. Although, minimizing the empirical risk can be easily attained since it is "measurable" from the observed data, we are not aware of the conditions under which a learning algorithm generalizes. The generalization risk is often impossible to determine since the true risk is unknown. However, there is a broad range of established upper bounds on $R_{gen}$ for both deterministic and randomized learning algorithms, and in both regression and classification cases. Initially introduced by Vapnik [Vapnik, 1999], generalization risk bounds have been derived on the basis of uniform convergence [Vapnik, 1999], algorithmic stability [Elisseeff *et al.*, 2003; Elisseeff *et al.*, 2005], generic chaining [Audibert and Bousquet, 2007; Talagrand, 1996], the PAC-Bayesian framework [Audibert and Bousquet, 2007; McAllester, 2003], Rademacher and Gaussian complexities [Bartlett and Mendelson, 2002], and robustness-based analysis [Xu and Mannor, 2012]. Some of these bounds are derived on the hypotheses selected by learning algorithms, while others bound the risks of the learning algorithms. For instance, the Vapnik-Chervonenkis (VC) theory [Blumer *et al.*, 1989] provides generalization bounds on hypotheses' risks, while stability bounds [Bousquet and Elisseeff, 2002; Kutin and Niyogi, 2002; Poggio *et al.*, 2004] are derived on learning algorithms' risks. For the theoretical justification in this study, we will focus on the latter.

Inspired by this insight, we designed a bias-variance balancing objective function that aims at tightening an algorithmic stability-based upper bound on the expected true risk in order to yield improved predictive performance. It utilizes distance correlation [Székely *et al.*, 2007; Székely *et al.*, 2009], a measure for statistical dependence, to control mutual stability between a model's loss w.r.t. given data and the data itself. This enables learning from empirical evidence, while at the same time enhancing the overall model's generalization performance. Since ensemble construction is a natural way of achieving greater generalization performance, we present the GeneraLization-Aware Collaborative Ensemble Regressor (GLACER), a collaborative ensemble-based model for structured regression that optimizes a generalization-aware objective function. GLACER bags multiple graphical models, namely Gaussian CRFs, and allows them to interact by exchanging examples in a way that minimizes the proposed objective function. Here, we utilize the idea of exchanging examples between ensemble components in a structured regression setting. This decreases GLACER's empirical risk and at the same time accounts for its generalization risk by tightening an upper bound of its expected true risk.

GLACER has been assessed on both synthetic and real-world data, and compared against both traditional and structured regression models. In the conducted experiments, GLACER's predictions have shown to be stable, yielding statistically significant improvements in average MSE.

GLACER's underlying framework can be viewed as a collaborative ensemble that aims to minimize a generalization error bound. Thus, it can be extended beyond regression by replacing its base components by any kind of learning models (e.g. binary classification models [Arsov *et al.*, 2017]).

*The main contributions* of this work are summarized as follows:
1) Theory and calculation of a *stability-based generalization error bound* are bridged by leveraging the *distance correlation* measure;
2) A *bias-variance balancing objective* function that takes advantage of distance correlation is designed to indirectly control the mutual stability between a regression model's loss on a dataset and the dataset itself, and thus to effectively *address the trade-off between underfitting and overfitting*;
3) A *collaborative* ensemble regressor was developed whose components are allowed to interact through a novel *example-exchange-driven optimization* in order to optimize the proposed objective.

## 2 Problem Formulation

Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be an observation space of input-output pairs, where $\mathcal{X} \in \mathbb{R}^d$ and $\mathcal{Y} \in \mathbb{R}$ denote the input and output space, respectively. Further, let $\mathcal{D} = \{z_1 = (\mathbf{x}_1, y_1), \ldots, z_N = (\mathbf{x}_N, y_N)\} \in \mathcal{Z}^N$ be a training set of $N$ input-output pairs $(\mathbf{x}_i, y_i)$ referred to as examples, where $\mathcal{Z}^N$ represents the space of all training sets of size $N$. The task is the prediction of an unobserved vector $\mathbf{y}' = [y'_1, \ldots, y'_{N'}]^\top$ of real-valued outputs (targets), given their corresponding inputs $\mathbf{X}' = [\mathbf{x}'_1, \ldots, \mathbf{x}'_{N'}]^\top$.

## 3 Main Theoretical Insight

The proposed objective function optimized by GLACER (introduced later in Section 4.2) relies on the following insights:

- An upper bound, derived on the expected true risk $\hat{R}_{true}$, i.e. the true risk of any learning algorithm $\mathcal{L}$:

$$\hat{R}_{true}(\mathcal{L}) \leq \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{h|\mathcal{D}}[R_{emp}(h, \mathcal{D})]] + 1 - \mathcal{S}(\ell(\cdot, h), z_{trn}), \tag{1}$$

where $R_{emp}(h, \mathcal{D})$ denotes the empirical risk of a hypothesis $h$ (inferred by $\mathcal{L}$), while $\mathcal{S}(\ell(\cdot, h), z_{trn})$ represents the mutual stability between the loss of $h$ and a random training example $z_{trn}$.

- A proposed bias-variance balancing objective function designed to minimize the upper bound in Eq. (1):

$$R_{obj}(h, \mathcal{D}) = \sqrt{R_{emp}(h, \mathcal{D})^2 + dCorr(\ell(\cdot, h), z_{trn})^2}, \tag{2}$$

where $R_{emp}(h, \mathcal{D})$ is the empirical risk of $h$ and $dCorr(\ell(\cdot, h), z_{trn})$ is the distance correlation, a measure of statistical dependence between $\ell(\cdot, h)$ and $z_{trn}$.
Next, we discuss the derivations of Eqs. (1) and (2) in detail.

### 3.1 Theoretical Background

The main goal is to address the fundamental learning problem of balancing between the notions of underfitting and overfitting. The former is controlled by the empirical risk $R_{emp}$, while maintaining a low true risk $R_{true}$ assists in avoiding the latter. A convenient way to take both into account is by analyzing their absolute difference, i.e. the generalization risk $R_{gen} = |R_{emp} - R_{true}|$. However, a low $R_{gen}$ guarantees similar risks, leaving open the possibility of similar but still high risks. Therefore, we consider maintaining a low empirical risk $R_{emp}$ while minimizing the risk difference $R_{gen}$.

The empirical risk $R_{emp}$ can be further minimized since it is "measurable", i.e. its value can be calculated with respect to a particular loss function using the observed data. However, the true risk $R_{true}$ is unknown. Therefore, the risk difference $R_{gen}$ cannot be directly calculated. Instead, one can try to minimize the value of the upper bound of $R_{gen}$. There is a broad range of upper bounds on $R_{gen}$, but their values are often not directly controllable. In this work, we focus on the bounds of the *expected generalization risk* $\hat{R}_{gen}$ which were recently established and are based on the notion of stability.

Here, we define the generalization risk of a learning algorithm $\mathcal{L} : \cup_{N=1}^{\infty} \mathcal{Z}^N \rightarrow \mathcal{H}$, or simply the expected generalization risk $\hat{R}_{gen}$. The algorithm $\mathcal{L}$ selects a hypothesis $h : \cup_{N=1}^{\infty} \mathcal{Z}^N \rightarrow \mathcal{Y}$ from a hypothesis space $\mathcal{H}$ using a subset of the whole observation space. Given a bounded parametric loss function $\ell(\cdot, h) : \mathcal{Z} \rightarrow [0, 1]$, the generalization risk of $\mathcal{L}$ w.r.t. $\ell(\cdot, h)$ is defined as the absolute difference between the empirical and true risks of $\mathcal{L}$, i.e.

$$\hat{R}_{gen}(\mathcal{L}) = |\hat{R}_{emp}(\mathcal{L}) - \hat{R}_{true}(\mathcal{L})|.$$

The empirical and true risks of $\mathcal{L}$ are defined as the expected empirical and expected true risks of $h$, i.e.

$$\begin{aligned} \hat{R}_{emp}(\mathcal{L}) &= \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{h|\mathcal{D}}[R_{emp}(h, \mathcal{D})]]; \\ \hat{R}_{true}(\mathcal{L}) &= \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{h|\mathcal{D}}[R_{true}(h)]], \end{aligned} \tag{3}$$

where $R_{emp}(h, \mathcal{D}) = \frac{1}{N} \sum_{i=1}^{N} \ell(z_i, h)$; $R_{true}(h) = \mathbb{E}_{z \sim \mathbb{P}(z)}[\ell(z, h)]$. We next present a theorem that outlines a stability-based bound of $\hat{R}_{gen}$ which the proposed objective function is based on.

**Theorem 1 ([Alabdulmohsin, 2017; Alabdulmohsin, 2015])** *For any learning algorithm $\mathcal{L} : \cup_{N=1}^{\infty} \mathcal{Z}^N \rightarrow \mathcal{H}$, algorithmic stability is both a necessary and sufficient condition for uniform generalization. Moreover,*

$$\hat{R}_{gen} = |\hat{R}_{true}(\mathcal{L}) - \hat{R}_{emp}(\mathcal{L})| \leq 1 - \mathcal{S}(\ell(\cdot, h), z_{trn}), \tag{4}$$

where $\mathcal{S}(\ell(\cdot, h), z_{trn})$ denotes the mutual stability between $\ell(\cdot, h)$ and $z_{trn}$ which essentially represents the overlap between their probability distributions. For the complete proof of the theorem, just follow the reference to its original source.

Next, we justify our choice of the upper bound in Eq. (4) by providing the main two reasons for it:

- In [Alabdulmohsin, 2015] it has been proven that uniform generalization is essentially equivalent to algorithmic stability. Since algorithmic stability is defined by the mutual stability $\mathcal{S}(\ell(\cdot, h), z_{trn})$ (on which the bound Eq. (4) is based), it follows that mutual stability is also connected to uniform generalization. In other words, one needs to be able to control mutual stability in order to improve generalization performance.

- The value of the chosen bound still cannot be directly minimized, but at least it can be indirectly controlled, unlike most of the bounds on $R_{gen}$.

We focus solely on $1 - \mathcal{S}(\ell(\cdot, h), z_{trn})$ as it is the tightest bound presented in [Alabdulmohsin, 2015, Trm. 1].

### 3.2 Bias-Variance Balancing Objective Function

The ultimate goal is to minimize both $\hat{R}_{emp}$ and the upper bound of $\hat{R}_{gen}$ at the same time, which can be achieved by tightening the upper bound of $\hat{R}_{true}$. Expressing the chosen bound (Eq. (4)) in terms of $\hat{R}_{true}$ yields

$$\hat{R}_{true}(\mathcal{L}) \leq \hat{R}_{emp}(\mathcal{L}) + 1 - \mathcal{S}(\ell(\cdot, h), z_{trn}). \tag{5}$$

According to Eq. (3), the upper bound in Eq. (5) can be rewritten as

$$\hat{R}_{true}(\mathcal{L}) \leq \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{h|\mathcal{D}}[R_{emp}(h, \mathcal{D})]] + 1 - \mathcal{S}(\ell(\cdot, h), z_{trn}). \tag{6}$$

To control the above upper bound, we define the following objective function:

$$R_{obj}(h, \mathcal{D}) = \sqrt{R_{emp}(h, \mathcal{D})^2 + dCorr(\ell(\cdot, h), z_{trn})^2}. \tag{7}$$

It is clear that the first term in Eq. (7), which minimizes the empirical error $R_{emp}(h, \mathcal{D})$, will induce a decrease of $\hat{R}_{emp}(\mathcal{L})$. The second term $dCorr(\ell(\cdot, h), z_{trn})$, defined later in Eq. (8), measures the statistical dependence between the loss $\ell(\cdot, h)$ and a random training example $z_{trn}$. Minimizing $dCorr$ tightens the upper bound of $\hat{R}_{gen}$.

In the following section, we provide details on the connection between the notions of mutual stability and statistical dependence, and further explain how distance correlation can indirectly control the stability term $\mathcal{S}(\ell(\cdot, h), z_{trn})$ in Eq. (6).

### 3.3 How Statistical Dependence Controls Stability?

From an information-theoretic perspective, as the variational information $\mathcal{I}(\ell(\cdot, h), z_{trn}) = 1 - \mathcal{S}(\ell(\cdot, h), z_{trn})$ decreases, the empirical loss $\ell(z_i, h)$ becomes more representative to the true loss $\ell(z, h)$, $z \sim \mathbb{P}(z)$. In terms of stability theory, this means that the learning algorithm that selected $h$ is becoming more stable, since a high value of $\mathcal{S}(\ell(\cdot, h), z_{trn})$ indicates that the probability distribution of $\ell(\cdot, h)$ is not perturbed by a random training example $z_{trn}$. Perfect mutual stability $\mathcal{S}(\ell(\cdot, h), z_{trn}) = 1$ is achieved when $\mathcal{I}(\ell(\cdot, h), z_{trn}) = 0$, i.e. when $\ell(\cdot, h)$ and $z_{trn}$ are statistically independent. Therefore, we account for the dependence between them by introducing the second term in our objective function, which relies on the distance correlation measure between $\ell(\cdot, h)$ and $z_{trn}$. Thinking of $\ell(\cdot, h)$ and $z_{trn}$ as random vectors $\mathbf{L}$ and $\mathbf{Z}$ whose observations are $\ell(z_1, h), \ldots, \ell(z_N, h)$ and $z_1, \ldots, z_N$, respectively, the (empirical) distance correlation between them is defined as the square root of

$$dCorr_N^2(\mathbf{L}, \mathbf{Z}) = \frac{dCov_N^2(\mathbf{L}, \mathbf{Z})}{\sqrt{dCov_N^2(\mathbf{L}, \mathbf{L})\; dCov_N^2(\mathbf{Z}, \mathbf{Z})}}, \tag{8}$$

when $dCov_N^2(\mathbf{L}, \mathbf{L}) dCov_N^2(\mathbf{Z}, \mathbf{Z}) > 0$. Otherwise, $dCorr_N^2(\mathbf{L}, \mathbf{Z}) = 0$. In Eq. (8), the squared sample distance covariance is calculated as

$$dCov_N^2(\mathbf{L}, \mathbf{Z}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (d_{ij}^{\mathbf{L}} - d_{i\cdot}^{\mathbf{L}} - d_{\cdot j}^{\mathbf{L}} + d_{\cdot\cdot}^{\mathbf{L}}) \\ (d_{ij}^{\mathbf{Z}} - d_{i\cdot}^{\mathbf{Z}} - d_{\cdot j}^{\mathbf{Z}} + d_{\cdot\cdot}^{\mathbf{Z}}).$$

Here, $d_{i\cdot} = \frac{1}{N}\sum_{j=1}^{N} d_{ij}$, $d_{\cdot j} = \frac{1}{N}\sum_{i=1}^{N} d_{ij}$, and $d_{\cdot\cdot} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}$, where $d_{ij}$ is the euclidean distance between the $i$-th and $j$-th observations of the random vector for which it is being calculated.

Distance correlation generalizes the idea of correlation between $\mathbf{L}$ and $\mathbf{Z}$, while preserving the following properties:

(i) $dCorr(\mathbf{L}, \mathbf{Z})$ is defined for two random vectors of arbitrary, not necessarily equal dimensions;
(ii) $dCorr(\mathbf{L}, \mathbf{Z}) = 0$ iff $\mathbf{L}$ and $\mathbf{Z}$ are independent;
(iii) $0 \leq dCorr(\mathbf{L}, \mathbf{Z}) \leq 1$.

From (ii), it is clear that minimizing $dCorr(\mathbf{L}, \mathbf{Z})$ in the objective's second term may cause a positive change in the mutual stability $\mathcal{S}(\ell(\cdot, h), z_{trn})$. The assumption here is that, in the idealistic case, when the value of $dCorr(\mathbf{L}, \mathbf{Z})$ reaches zero, $\mathbf{L}$ and $\mathbf{Z}$ are independent and $1 - \mathcal{S}(\ell(\cdot, h), z_{trn})$ vanishes in the upper bound (Eq. (6)). A direct relation between distance correlation and information-theoretic mutual stability is not claimed, but rather, in this case it is *inspired* by (*indirectly* related to) the concept of mutual stability.

## 4 Methodology

### 4.1 Preliminaries: Gaussian CRF

A Continuous Conditional Random Field (CCRF) [Qin *et al.*, 2009] models the conditional distribution of the outputs $\mathbf{y}$, given all inputs $\mathbf{X}$, as $P(\mathbf{y}|\mathbf{X}) = \frac{1}{Z}\exp\left\{-\alpha \sum_{i=1}^{N} (y_i - \phi(\mathbf{x}_i))^2 - \beta \sum_{i \sim j} S_{ij} (y_i - y_j)^2\right\}$, where $[\alpha, \beta]^\top$ is a parameter vector, while $Z$ is a normalization constant, calculated as an integral over $\mathbf{y}$ of the term in the exponent. The first term in the exponent models the relevance of an "unstructured predictor" $\phi$ by assigning a weight $\alpha$. In the second term, $S_{ij}$ is extracted by a user-defined similarity measure $s(\mathbf{x}_i, \mathbf{x}_j)$ assuming that if the two inputs $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar, their corresponding outputs $y_i$ and $y_j$ are also similar. The similarity matrix $\mathbf{S} = [S_{ij}]_{N \times N}$ can be seen as an adjacency matrix of an undirected weighted graph whose relevance is determined by the $\beta$ parameter.

**Structured Learning.** Since the constituents of the exponent in $P(\mathbf{y}|\mathbf{X})$ are defined as quadratic functions in terms of $\mathbf{y}$, the conditional probability can be transposed directly onto a multivariate Gaussian distribution, hence the CCRF becomes a Gaussian CRF (GCRF) [Radosavljevic *et al.*, 2010]. Learning a GCRF boils down to determining the precision matrix $\mathbf{Q} = \alpha\mathbf{I} + \beta\mathbf{L}$, where $\mathbf{L}$ is the Laplacian of $\mathbf{S}$ and $\mathbf{I}$ is an identity matrix. Note that $\mathbf{Q}$ is used to obtain an explicit expression for the inverse covariance matrix $\mathbf{\Sigma}^{-1} = 2\mathbf{Q}$. The learning is governed by a convex optimization that strives to determine $[\hat{\alpha}, \hat{\beta}]^\top = \arg\max_{\alpha,\beta} \log(P(\mathbf{y}|\mathbf{X}; \alpha, \beta))$.

**Inference.** Since the model relies on a multivariate Gaussian distribution, the estimate of a target vector $\mathbf{y}'$ is the distribution's expected value, for which $P(\mathbf{y}'|\mathbf{X}')$ is maximized. Hence, the inference task comes down to calculating $\boldsymbol{\mu} = \boldsymbol{\Sigma}\mathbf{b} = \alpha\mathbf{Q}^{-1}[\phi(\mathbf{x}'_1), \ldots, \phi(\mathbf{x}'_{N'})]^\top$.

### 4.2 GeneraLization-Aware Collaborative Ensemble Regressor (GLACER)

The proposed model, GLACER, represents a collaborative ensemble model for structured regression that strives to avoid both underfitting and overfitting by optimizing the proposed objective function (Eq. (7)). Since bagging [Andonova *et al.*, 2002] combined with boosting [Friedman, 2001] can be very effective for achieving both variance and bias reduction [Büchlmann and Yu, 2002], GLACER sub-bags multiple GCRFs, each utilizing an LSBoost model as an unstructured predictor. Moreover, it employs collaboration between its constituents by allowing them to interactively exchange examples during training. This example-exchange-guided optimization aims to minimize Eq. (7), thus enhancing the overall model's predictive performance. GLACER's training procedure is outlined in Algorithm 1.

**Ensemble Construction** (Lines 1-5 in Algorithm 1). In order to employ multiple "local" GCRF models to discover different data substructures, a training set $\mathcal{D}$ and its corresponding similarity matrix $\mathbf{S}$ are sampled uniformly $M$ times without replacement, thus generating $M$ data subsets $\mathcal{D}^1, \ldots, \mathcal{D}^M$ of size $\eta N$, where $\eta \in (0, 1)$, and corresponding $M$ similarity submatrices $\mathbf{S}^1, \ldots, \mathbf{S}^M$. Then, each $\mathcal{D}^m$ is used to train a single GCRF component $F_{\mathcal{D}^m}$ characterized by $\alpha^m$ and $\beta^m, \forall m = 1, \ldots, M$. Since it is obvious that $\mathbf{S}^m$ is passed as an input to the $m$-th GCRF component together with $\mathcal{D}^m$, $\mathbf{S}^m$ is left out from the subscript of $F_{\mathcal{D}^m}$ for simplicity of notation. The final predictions for the output values are made by combining all GCRF components' outcomes in a subbagging fashion: $\Phi_{\mathcal{D}}(\mathbf{X}, \mathbf{S}) = \frac{1}{M}\sum_{m=1}^{M} F_{\mathcal{D}^m}(\mathbf{X}, \mathbf{S})$.

Next, we proceed by presenting a general interactive optimization procedure that determines which examples should fall in which subsets, that is, modify $\mathcal{D}^m$ and $\mathbf{S}^m$ for the components $F_{\mathcal{D}^m}$ so as to decrease the objective function $R_{obj}$.

**Step I** (Lines 7-17 in Algorithm 1). In the first step of the optimization, candidate (potential) swaps between GCRF components are made; and the decrease in the objective function is calculated after each of them occurs. Examples are tentatively assorted for exchange, and selection is determined by evaluating the losses of each GCRF component with respect to all examples in its subset. When evaluating the loss of a regression model, the coefficient of determination $R^2$ is often used. The range of $R^2$ values is $(-\infty, 1]$. Since there is a constraint related to the loss function which should fall between 0 and 1, we define a loss function that measures how well a GCRF component $F_{\mathcal{D}^m}$ fits an example $z$ as follows,

$$\begin{aligned}\ell(z, F_{\mathcal{D}^m}) &= 1 - e^{r(z, F_{\mathcal{D}^m})-1} \\ &= 1 - \exp\left\{\left(\frac{1}{N} - \frac{(y - F_{\mathcal{D}^m}(\mathbf{x}))^2}{Var(\mathbf{y}^m)}\right) - 1\right\}, \quad z \in \mathcal{D}^m.\end{aligned}$$

Thereafter, the worst-fit example $z_*^m$ of the $m$-th GCRF component is determined by $z_*^m = \arg\max_{z \in \mathcal{D}^m} \ell(F_{\mathcal{D}^m}, z), \quad \forall m = 1, \ldots, M$. In order to improve the predictive performance of a single GCRF component, the worst-fit example from its training subset is exchanged with another worst-fit example from another GCRF component's subset. To avoid duplicating examples and compromising the subset size equality among the components, a pair of GCRF components is allowed to exchange examples only if each of them receives an example from the other one that is not present in its subset, and vice versa. Therefore, after an example exchange occurs, $\Phi_{\mathcal{D}}$ will still represent a valid subbagging ensemble. As per this constraint, the max-loss examples are exchanged between a pair of GCRF components as $(\mathcal{D}_p^j, \mathcal{D}_p^k) = \left(\mathcal{D}^j \setminus \{z_*^j\} \cup \{z_*^k\}, \mathcal{D}^k \setminus \{z_*^k\} \cup \{z_*^j\}\right)$, if $z_*^k \notin \mathcal{D}^j \wedge z_*^j \notin \mathcal{D}^k$, where $\mathcal{D}_p^j$ and $\mathcal{D}_p^k$ denote the modified versions of $\mathcal{D}^j$ and $\mathcal{D}^k$, respectively. Otherwise, the original subsets remain the same, i.e. $(\mathcal{D}_p^j, \mathcal{D}_p^k) = (\mathcal{D}^j, \mathcal{D}^k)$.

The exchange of $z_*^j$ and $z_*^k$ is referred to as one "candidate swap". It should be noted that, along with the temporary update of the $j$-th and $k$-th subsets, their corresponding similarity matrices $\mathbf{S}^j$ and $\mathbf{S}^k$ are also updated accordingly. After performing this candidate swap, the outcome of the overall model for arbitrary $\mathbf{X}$ and $\mathbf{S}$ is

$$\begin{aligned}&\Phi_{\mathcal{D}(j,k)}(\mathbf{X}, \mathbf{S}) \\ &= \frac{1}{M}\Big(F_{\mathcal{D}_p^j}(\mathbf{X}, \mathbf{S}) + F_{\mathcal{D}_p^k}(\mathbf{X}, \mathbf{S}) + \sum_{\substack{m=1 \\ m \neq j,k}}^{M} F_{\mathcal{D}^m}(\mathbf{X}, \mathbf{S})\Big).\end{aligned}$$

To quantify the contribution of each swap, the difference between the values of the objective function $R_{obj}$ (Eq. (7)), before and after a swap occurs, is calculated:

$$\Delta_{jk} = R_{obj}(\Phi_{\mathcal{D}}, \mathcal{D}) - R_{obj}(\Phi_{\mathcal{D}(j,k)}, \mathcal{D}).$$

The value of $\Delta_{jk}$ essentially represents the improvement in the objective function value brought by the collaboration between the $j$-th and $k$-th GCRF components. This measure is calculated for all pairs of GCRF components.

**Step II** (Lines 18-20 in Algorithm 1). Once all tentative candidate swaps are examined, examples are exchanged between those two GCRF components that foster the highest decrease in the objective function.

**Convergence** (Line 6 in Algorithm 1). The aforedescribed steps constitute one iteration of GLACER's training procedure and all of them are repeated until no additional exchange between any pair of GCRF components can further decrease the objective function.

**Monotonicity of $R_{obj}$.** Let us assume that $\tau$ is the current optimization iteration. If convergence is not the case at this iteration, then there must exist at least one pair $(j, k)$ for which $\Delta_{jk} > 0$. Henceforth, an exchange between a pair of GCRF components must occur, and the fact that $\Delta_{jk} > 0$ only if the exchange fosters an improvement in the value of $R_{obj}$ guarantees that $R_{obj}$ will monotonically decrease as $\tau$ increments by one.

**Algorithm 1** GLACER

```
Input: Training set $\mathcal{D}$, #components $M$,
       sub-sampling fraction $\eta$, similarity matrix $\mathbf{S}$
Procedure:
for $m = 1, \dots, M$ do
    $(\mathcal{D}^m, \mathbf{S}^m) \leftarrow Sub\text{-}sample(\mathcal{D}, \mathbf{S}, \eta)$
    $\phi \leftarrow Train\_LSBoost(\mathcal{D}^m)$
    $F_{\mathcal{D}^m} \leftarrow Train\_GCRF(\mathcal{D}^m, \mathbf{S}^m, [\phi(\mathbf{x})]^{\top}_{\mathbf{x}\in\mathcal{D}^m})$
Construct $\Phi_{\mathcal{D}} = 1/M \sum_{m=1}^{M} F_{\mathcal{D}^m}(\mathbf{X}, \mathbf{S})$
while at least one $\Delta_{jk} > 0$ do
    for $m = 1, \dots, M$ do
        $z_*^m = \arg\max_{z\in\mathcal{D}^m} \ell(F_{\mathcal{D}^m}, z)$
    for all unique pairs $(j, k) \in [1, M]^2$ do
        if $z_*^k \notin \mathcal{D}^j \wedge z_*^j \notin \mathcal{D}^k$ then
            Calculate the objective $R_{obj}(\Phi_{\mathcal{D}}, \mathcal{D})$
            Swap examples between the $\mathcal{D}^j$ and $\mathcal{D}^k$
            Retrain $F_{\mathcal{D}^j}$ and $F_{\mathcal{D}^k}$
            Construct a modified ensemble $\Phi_{\mathcal{D}^{(j,k)}}$
            Calculate $R_{obj}(\Phi_{\mathcal{D}^{(j,k)}}, \mathcal{D})$
            $\Delta_{jk} = R_{obj}(\Phi_{\mathcal{D}}, \mathcal{D}) - R_{obj}(\Phi_{\mathcal{D}^{(j,k)}}, \mathcal{D})$
            Put $z_*^j$ and $z_*^k$ in their original subsets
    Find the optimal pair $(j^*, k^*) = \arg\max_{(j,k)} \Delta_{jk}$
    Permanently modify subsets $\mathcal{D}^{j^*}$ and $\mathcal{D}^{k^*}$
    In $\Phi_{\mathcal{D}}$: replace $F_{\mathcal{D}^j}$ by $F_{\mathcal{D}^{j^*}}$, and $F_{\mathcal{D}^k}$ by $F_{\mathcal{D}^{k^*}}$
Output: Return $\Phi_{\mathcal{D}}$
```

# 5 Experiments

## 5.1 Baselines

The baselines considered in this study are listed as follows:

- MLR: Multiple Linear Regression model.
- NN: Feed-Forward Neural Network with 3 layers.
- SVM: SVM regression model with an RBF kernel.
- SUBBAG: Variation of bagging that considers sampling at random, but without replacement to generate training subsets.
- RF: Random Forest ensemble that performs random sampling of examples to generate subsets, followed by a random feature selection within each subset.
- LSB: Gradient-boosting ensemble for additive expansions based on the least-squares fitting criterion.
- (C/NC)NL: Network Lasso, a structured regression model capable of simultaneous clustering and optimization on graphs. Both its convex (C)NL and non-convex (NC)NL variants were trained with $\lambda = 5$, same as in [Hallac *et al.*, 2015].

## 5.2 Setup

In all conducted experiments, the ensemble-based baselines (SUBBAG, RF, LSB) and GLACER were run with $M$ components, while the same sub-sampling fraction $\eta$ was used to construct training subsets for both SUBBAG and GLACER. Moreover, each baseline other than NL (which is already a structured approach) was run in both traditional (unstructured) and structured mode. A baseline's structured variant is obtained by passing it to a GCRF as its unstructured predictor. Structured variants are distinguished from unstructured ones by the "S-" at the beginning of their names (e.g. S-MLR denotes a structured MLR). Mean squared error (MSE) was calculated for all models. The average testing MSEs are reported, along with their corresponding two-sided confidence intervals at 90% confidence level.

## 5.3 Experiments on Synthetic Data

**Data Generation.** The synthetic data was generated such that it holds a certain structure ([Pavlovski *et al.*, 2017] used a similar generation process). First, $N = 3000$ examples $\mathbf{x}_i \in \mathbb{R}^d$ ($d = 5$) were generated such that each attribute is normally distributed according to a standard normal distribution. Thereafter, outputs were created as parameterized polynomials with uniformly distributed parameters. Normally distributed noise was applied to these outputs, yielding $\widetilde{\mathbf{y}}$. The structure among the examples was created on the basis of an Erdős-Rényi random graph $G$, each node corresponding to one example. Accordingly, the between-example similarities were calculated as $S_{ij} = e^{-|\widetilde{\mathbf{y}}_i - \widetilde{\mathbf{y}}_j|}$ in case an edge between $i$ and $j$ exists in $G$. Otherwise, $S_{ij}$ was set to zero. Finally, GCRF was utilized in a generative manner in order to infer the final outputs as $\mathbf{y} = \alpha(\alpha\mathbf{I} + \beta\mathbf{L})^{-1}\widetilde{\mathbf{y}}$, where $\alpha = 1$ and $\beta = 5$ were chosen, thus making the structure more significant than the input-output relationship within the data.

**Parameter Analysis.** The predictive performance of GLACER was analyzed under different sets of parameters. Using the aforedescribed data generation procedure, 10 different training and independent test sets were generated. GLACER was then run on each train/test pair with $M = 5, 10, 30$ components, while for each value of $M$ the sub-sampling fraction $\eta$ varied within $\{0.3, 0.5, 0.7\}$. The testing MSEs regarding all these different parameter sets are left out due to lack of space. As expected, GLACER stabilizes and achieves greater performance as $M$ increases. When $M = 10$, GLACER already shows good generalization performance, and $\eta$ does not seem to play a crucial role. Therefore, in the following experiments we chose to run GLACER with $M = 10$ and set $\eta$ to 0.3 for efficiency.

**Generalization Capability.** Different fractions of data (10%, 50% and the entire training set) were used to further investigate GLACER's generalization performance. For each training data size, the average MSE over 10 repetitions was calculated for GLACER and all baselines. As shown in Table 1, GLACER yields the lowest MSEs under all three training data sizes. Although all models' MSEs decrease with the increased size of training data, GLACER sustains stable predictions when only 50% training data is available, which clearly indicates the generalization performance of GLACER.

**Influence of Distance Correlation.** To evaluate the influence of distance correlation in the objective function, GLACER's generalization performance was evaluated on different fractions of the same synthetic data in two cases: (1) when the $dCorr$ term is excluded from $R_{obj}$; (2) when $dCorr$ is incorporated in $R_{obj}$. According to Table 2, GLACER manifests lower average MSEs when $dCorr$ is used in $R_{obj}$. This is consistent as the training data increases. Besides, without $dCorr$, the average MSE worsens once the training set fraction increases from 50% to 100% which might be an indication of overfitting. On the contrary, incorporating

| **Frac.**<br>**Model** | **10%** | **50%** | **100%** |
|---|---|---|---|
| MLR | 2.91 ± 0.25 | 2.33 ± 0.05 | 2.40 ± 0.09 |
| S-MLR | 2.23 ± 0.86 | 1.51 ± 0.05 | 1.64 ± 0.06 |
| NN | 2.79 ± 1.32 | 1.42 ± 0.21 | 1.18 ± 0.14 |
| S-NN | 3.31 ± 2.25 | 0.95 ± 0.18 | 0.76 ± 0.09 |
| SVM | 3.60 ± 0.12 | 2.39 ± 0.12 | 2.46 ± 0.10 |
| S-SVM | 3.60 ± 0.10 | 1.77 ± 0.12 | 1.89 ± 0.10 |
| SUBBAG | 4.97 ± 0.38 | 1.25 ± 0.01 | 0.86 ± 0.03 |
| S-SUBBAG | 5.57 ± 0.45 | 0.91 ± 0.03 | 0.63 ± 0.04 |
| RF | 5.33 ± 0.48 | 1.63 ± 0.05 | 1.09 ± 0.04 |
| S-RF | 7.32 ± 0.55 | 1.35 ± 0.03 | 0.94 ± 0.04 |
| LSB | 3.47 ± 0.35 | 2.79 ± 0.05 | 2.65 ± 0.05 |
| S-LSB | 2.00 ± 0.20 | 0.90 ± 0.02 | 1.22 ± 0.11 |
| (C)NL | 2.58 ± 0.48 | 1.20 ± 0.04 | 0.64 ± 0.04 |
| (NC)NL | 2.53 ± 0.41 | 1.32 ± 0.05 | 0.97 ± 0.05 |
| **GLACER** | **0.72 ± 0.11** | **0.25 ± 0.01** | **0.25 ± 0.004** |

Table 1: Average testing MSE when 10%, 50%, and all training data is supplied.

| $R_{obj}$<br>**Frac.** | **Without** $dCorr$ | **With** $dCorr$ |
|---|---|---|
| **10%** | 0.740 ± 0.112 | 0.716 ± 0.112 |
| **50%** | 0.435 ± 0.007 | 0.245 ± 0.006 |
| **100%** | 0.529 ± 0.008 | 0.252 ± 0.004 |

Table 2: Average testing MSE, obtained before and after using $dCorr$ within $R_{obj}$.

$dCorr$ in $R_{obj}$ prevents from large increases in MSE. This indicates that $dCorr$ plays an important role in the model's generalization performance.

### 5.4 Real-World Datasets

**Sacramento Real-Estate.** A collection of 985 real estate transactions were observed in the Greater Sacramento area, California, made over a period of one week in May 2008. Each example refers to a house sale record that contains information about the number of bedrooms and bathrooms, the house area in square feet, and its location in terms of latitude and longitude. The regression task is to predict houses' prices based on their characteristics. Since some attribute values are missing, we used a dataset version that was pre-processed by [Hallac *et al.*, 2015]. An undirected similarity graph was constructed for each training and test set by coupling each house with its 5 nearest houses and the other way around. A weight $e^{-dist_{ij}}$ was assigned to each existing edge $(i, j)$ based on the geospatial distance between houses $i$ and $j$.

**Medicare Readmissions**[1]**.** This data consists of 1000 hospital records referring to hospitals that have more than ∼150 readmissions. Each record contains information about the number of discharges, the excess readmission ratio, as well as the estimated and expected readmission rates. Given a hospital record, the goal is to predict the number of readmissions at the hospital. The structure among the hospital records was constructed by calculating $dist_{ij}$ as the Manhattan distance between the attribute values of records $i$ and $j$, and assigning edge weights in the same way as for the Sacramento dataset.

| **Model** | **Sacramento** | **Medicare** |
|---|---|---|
| MLR | 0.507 ± 0.025 | 1755.708 ± 616.119 |
| S-MLR | 0.465 ± 0.024 | 525.551 ± 196.065 |
| NN | 0.516 ± 0.026 | 2037.421 ± 1199.805 |
| S-NN | 0.463 ± 0.023 | 1618.547 ± 1192.462 |
| SVM | 0.515 ± 0.031 | 1359.342 ± 697.91 |
| S-SVM | 0.479 ± 0.034 | 504.076 ± 221.228 |
| SUBBAG | 0.304 ± 0.017 | 441.524 ± 101.065 |
| S-SUBBAG | 0.262 ± 0.015 | 234.505 ± 74.378 |
| RF | 0.283 ± 0.02 | 508.294 ± 110.988 |
| S-RF | 0.249 ± 0.015 | 247.406 ± 35.814 |
| LSB | 0.288 ± 0.015 | 595.289 ± 136.174 |
| S-LSB | 0.25 ± 0.017 | 182.006 ± 24.919 |
| (C)NL | 0.368 ± 0.013 | 5012.614 ± 768.945 |
| (NC)NL | 0.38 ± 0.017 | 5012.614 ± 768.945 |
| **GLACER** | **0.225 ± 0.005** | **73.183 ± 9.032** |

Table 3: Average testing MSE obtained on real-world datasets.

**Results and Discussion.** The Sacramento Real-Estate dataset was split into a training set of 785 house transactions and a test set of 200 transactions (same as in [Hallac *et al.*, 2015]). As for Medicare Readmissions, half of the data was randomly sampled and used for training, while the other half was used for evaluation. The average testing MSEs obtained on both datasets are summarized in Table 3. From these results, it is evident that GLACER outperforms its alternatives by significant margins (the corresponding $p$-values are smaller than 0.01 for Sacramento, and 0.021 for Medicare). For instance, GLACER is 38.86% ($p$-value $= 5.6 \times 10^{-9}$) and 40.79% ($p$-value $= 2.3 \times 10^{-8}$) more accurate than the convex and non-convex NL on the Sacramento Real-Estate dataset, respectively. GLACER also manifests a lower average MSE than its building blocks, namely SUBBAG and S-LSB on both datasets. As for all other baselines, it obtains substantial MSE decreases on the Sacramento dataset ranging from 9.64% to 56.4%, and from 49.32% up to 96.41% on the Medicare dataset, thus demonstrating a considerable generalization capability.

In addition, GLACER is more stable compared to alternatives, as it has the tightest confidence interval for its average MSE. A possible explanation behind this would be the nature of GLACER's stability-encouraging objective function. Lastly, there is no overlap between the confidence interval for GLACER's average MSE and any other model's interval. Henceforth, GLACER's improvements are statistically significant.

## 6 Conclusion

In this work, we bridged theory and calculation of a stability-based generalization error bound by leveraging the distance correlation measure and proposed a bias-variance balancing objective function which utilizes the properties of this measure to address the trade-off between underfitting and overfitting. In addition, we introduced GLACER, a model for

[1] https://data.medicare.gov/data/hospital-compare

structured regression which optimizes the proposed objective function through an example-exchange-driven optimization. GLACER was assessed on multiple datasets, on which it manifested stable predictions and significantly outperformed a broad range of traditional and structured regression models.

## Acknowledgments

This research was supported in part by the Defense Advanced Research Projects Agency (DARPA) and the Army Research Office (ARO) under Contract no. DARPA-THOR-W911NF-16-C-0050, an IQVIA Big Data Analytics contract, NSF BIGDATA grant 14476570, and ONR/ONR Global grant N62909-16-1-2222.

## References

[Alabdulmohsin, 2015] Ibrahim M Alabdulmohsin. Algorithmic stability and uniform generalization. In *Advances in Neural Information Processing Systems*, pages 19–27, 2015.

[Alabdulmohsin, 2017] Ibrahim Alabdulmohsin. An information-theoretic route from generalization in expectation to generalization in probability. In *Artificial Intelligence and Statistics*, pages 92–100, 2017.

[Andonova *et al.*, 2002] Savina Andonova, Andre Elisseeff, Theodoros Evgeniou, and Massimiliano Pontil. A simple algorithm for learning stable machines. In *Proceedings of the 15th European Conference on Artificial Intelligence*, pages 513–517. IOS Press, 2002.

[Arsov *et al.*, 2017] Nino Arsov, Martin Pavlovski, Lasko Basnarkov, and Ljupco Kocarev. Generating highly accurate prediction hypotheses through collaborative ensemble learning. *Scientific Reports*, 7:44649, 2017.

[Audibert and Bousquet, 2007] Jean-Yves Audibert and Olivier Bousquet. Combining pac-bayesian and generic chaining bounds. *Journal of Machine Learning Research*, 8(Apr):863–889, 2007.

[Bartlett and Mendelson, 2002] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[Blumer *et al.*, 1989] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

[Bousquet and Elisseeff, 2002] Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2(Mar):499–526, 2002.

[Büchlmann and Yu, 2002] Peter Büchlmann and Bin Yu. Analyzing bagging. *Annals of Statistics*, pages 927–961, 2002.

[Elisseeff *et al.*, 2003] André Elisseeff, Massimiliano Pontil, et al. Leave-one-out error and stability of learning algorithms with applications. *NATO science series sub series iii computer and systems sciences*, 190:111–130, 2003.

[Elisseeff *et al.*, 2005] Andre Elisseeff, Theodoros Evgeniou, and Massimiliano Pontil. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6(Jan):55–79, 2005.

[Friedman, 2001] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[Hallac *et al.*, 2015] David Hallac, Jure Leskovec, and Stephen Boyd. Network lasso: Clustering and optimization in large graphs. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 387–396. ACM, 2015.

[Kutin and Niyogi, 2002] Samuel Kutin and Partha Niyogi. Almost-everywhere algorithmic stability and generalization error. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 275–282. Morgan Kaufmann Publishers Inc., 2002.

[McAllester, 2003] David A McAllester. Pac-bayesian stochastic model selection. *Machine Learning*, 51(1):5–21, 2003.

[Pavlovski *et al.*, 2017] Martin Pavlovski, Fang Zhou, Ivan Stojkovic, Ljupco Kocarev, and Zoran Obradovic. Adaptive skip-train structured regression for temporal networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017.

[Poggio *et al.*, 2004] Tomaso Poggio, Ryan Rifkin, Sayan Mukherjee, and Partha Niyogi. General conditions for predictivity in learning theory. *Nature*, 428(6981):419, 2004.

[Qin *et al.*, 2009] Tao Qin, Tie-Yan Liu, Xu-Dong Zhang, De-Sheng Wang, and Hang Li. Global ranking using continuous conditional random fields. In *Advances in neural information processing systems*, pages 1281–1288, 2009.

[Radosavljevic *et al.*, 2010] Vladan Radosavljevic, Slobodan Vucetic, and Zoran Obradovic. Continuous conditional random fields for regression in remote sensing. In *ECAI*, pages 809–814, 2010.

[Székely *et al.*, 2007] Gábor J Székely, Maria L Rizzo, Nail K Bakirov, et al. Measuring and testing dependence by correlation of distances. *The annals of statistics*, 35(6):2769–2794, 2007.

[Székely *et al.*, 2009] Gábor J Székely, Maria L Rizzo, et al. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.

[Talagrand, 1996] Michel Talagrand. Majorizing measures: the generic chaining. *The Annals of Probability*, pages 1049–1103, 1996.

[Vapnik, 1999] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[Xu and Mannor, 2012] Huan Xu and Shie Mannor. Robustness and generalization. *Machine learning*, 86(3):391–423, 2012.