2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Regression Ensemble with PSO Algorithms Based Fuzzy Integral

James N. K. Liu
Department of Computing
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
Email: james.liu@polyu.edu.hk

Yulin He
College of Mathematics and Computer Science
Hebei University
Baoding 071002, Hebei, China
Email: yulinhe@ieee.org

Yanxing Hu
Department of Computing
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
Email: csyhu@comp.polyu.edu.hk

Xizhao Wang
College of Mathematics and Computer Science
Hebei University
Baoding 071002, Hebei, China
Email: xizhaowang@ieee.org

***Abstract*—Similar to the ensemble learning for classification, regression ensemble also tries to improve the prediction accuracy through combining several "weak" estimators which are usually high-variance and thus unstable. In this paper, we propose a new scheme of fusing the weak *Priestley-Chao Kernel Estimators* (PCKEs) based on Choquet fuzzy integral, which differs from all the existing models of regressor fusion. The new scheme uses Choquet fuzzy integral to fuse several target outputs from different PCKEs, in which the optimal bandwidths are obtained with cross-validation criteria. The key of applying fuzzy integral to PCKE fusion is the determination of fuzzy measure. Considering the advantage of particle swarm optimization (PSO) algorithm on convergence rate, we use three different PSO algorithms, i.e., standard PSO (SPSO), Gaussian PSO (GPSO) and GPSO with Gaussian jump (GPSOGJ), to determine the general and λ fuzzy measures. The finally experimental results on 6 standard testing functions show that the new paradigm for regression ensemble based on fuzzy integral is more accurate and stable in comparison with any individual PCKE. This demonstrates the feasibility and effectiveness of our proposed regression ensemble model.**

I. Introduction

Ensemble learning [1] [2] is a fusion strategy which tries to make the final decision by integrating the multiple feedbacks from different base-learners so as to reduce the decision maker's variance and improve its robustness and accuracy. That is to say a strong learner will be produced by organizing some weak ones in a proper way. Commonly, these weak learners are integrated through the majority voting for classification and a linear combination for regression [3]. In recent years, the ensemble learning for classification has been well studied. There are a number of classical works which introduce the ensemble strategies for different classifiers, e.g., boosting or bagging based ensembles for decision trees [4], neural networks [5] and support vector machines [6], etc. However, just as Moreiraa, et al. said in [7], the successful ensemble learning approaches for classification techniques are often not directly applicable to regression. Thus, unlike the sophisticated ensemble methods for classification, the regression ensemble often uses the weighted or ordered weighted average of base-learners to conduct the prediction, where several different methods are developed to determine the weights [7].

The weighted average and ordered weighted average operators are good choices to deal with the different importance of individual base-regressor, but these two methods are under an assumption that interaction does not exist among the individual regressors. However, this assumption may not be true in many real problems. If the interaction is considered, fuzzy integrals [8] [9] may be a better choice. The fuzzy integral as a fusion tool, in which the non-additive measure can clearly express the interaction among regressors, and the importance of each individual regressor has its particular advantages. Motivated by the definition of fuzzy integral which can be considered as a mechanism of maximizing the potential efficiency of base-regressor, we construct a new approach for regression ensemble based on fuzzy integral in this paper.

One difficulty for applying fuzzy integrals in regressor fusion is how to determine the fuzzy measures. There are some methods to determine fuzzy measures such as gradient descent (GD) [10], genetic algorithm (GA) [11], neural network (NN) [12], etc. Although using GD, GA and NN to determine the fuzzy measures is successful to some extent, there exist many limitations in the application process. For example, GD and NN frequently fall into the local minimum, and GA is much slower. It is necessary to mine new computational techniques for determining fuzzy measures. In 2011, Wang, et al. [13] proposed particle swarm optimization (PSO) [14] based fuzzy measure determination. The theoretical analysis and experimental comparison demonstrated the superior performance of PSO based methods. Thus, we use PSO to determine the fuzzy measure in fuzzy integral based regression ensemble in this paper. The main contributions of this paper can be summarized as 1) using fuzzy integral to construct the regression ensemble and 2) applying three different PSOs (i.e., standard PSO-SPSO [14], Gaussian PSO-GPSO [15] and GPSO with Gaussian jump-GPSOGJ [16]) to determine the general and λ fuzzy measures. In our study, a kind of kernel regressor, i.e., Priestley-Chao kernel estimator-PCKE [17], is selected as the base-regressor. We call the kernel regression ensemble model based on fuzzy integrals with the general and λ fuzzy measures $\mathrm{KREFI_g}$ and KREFI_λ respectively.

The rest of this paper is organized as follows: Section II introduces some basic concepts of fuzzy measure and fuzzy integral. The new regression ensemble model-KREFI is presented in Section III. Some simulations are performed in Section IV to show KREFI's effectiveness and efficiency. Finally, Section V concludes the paper.

II. Fuzzy Measure and Fuzzy Integral

A. Fuzzy Measure

Definition 1 [13]: Let X be a finite set and 2^X be the power set of X. If set function $\mu : 2^X \to [0,1]$ satisfies the following conditions:

1) $\mu(\emptyset) = 0$ and $\mu(X) = 1$;
2) If $E \subset 2^X$, $G \subset 2^X$ and $E \subset G$, then $\mu(E) \leq \mu(G)$,

then μ is called a general fuzzy measure defined on 2^X.

From Definition 1, we can find that there are total $2^k - 2$ measure values which are needed to determine for general fuzzy measure if the set X has k elements. When k is a very large number, the computational complexity of determining fuzzy measure is higher. In order to decrease this high complexity, some special fuzzy measure is proposed, e.g., λ fuzzy measure.

Definition 2 [13]: Let X be a finite set and 2^X be the power set of X. If set function $\mu : 2^X \to [0,1]$ satisfies the following conditions:

1) $\mu(\emptyset) = 0$ and $\mu(X) = 1$;
2) $\mu(A \cup B) = \mu(A) + \mu(B) + \lambda \times \mu(A) \times \mu(B)$, $\forall A, B \subset X$, $A \cap B = \emptyset$, $\lambda \in (-1, +\infty)$,

then μ is called a λ fuzzy measure defined on 2^X, where the parameter λ can be solved according to Theorem 1:

Theorem 1 [13]: The parameter λ of λ fuzzy measure can be calculated by Eq (1):

$$\prod_{j=1}^{k} (1 + \lambda \mu_j) = 1 + \lambda, \tag{1}$$

where $\mu_j = \mu(\{x_j\})$ is the measure value of singleton set $\{x_j\}$, let $X = \{x_1, x_2, \cdots, x_k\}$. [13] proves that it is only one λ meeting $\lambda > -1$ that can be solved from Eq. (1).

When we know the measure values on k singleton sets and λ, the measure values on other subsets of X can be determined as the following Theorem 2:

Theorem 2 [13]: For $\forall E \subset X$, its λ fuzzy measure is

$$\mu(E) = \frac{1}{\lambda}\left[\prod_{x_j \in E} (1 + \lambda \mu_j) - 1\right]. \tag{2}$$

For λ fuzzy measure, there are only k measure values needed to be determined, i.e., $\mu(\{x_1\})$, $\mu(\{x_2\})$, $\cdots$, $\mu(\{x_k\})$.

B. Fuzzy Integral

In fact, fuzzy integral [18] [19] is a kind of generalized integral with respect to fuzzy measure. There are some different types of fuzzy integrals which have been suggested in the literature [20] [21]. In this paper, we only give the introduction of Choquet integral on which our regression ensemble models $\mathrm{KREFI_g}$ and KREFI_λ are based.

Definition 3 [13]: For the finite set $X = \{x_1, x_2, \cdots, x_k\}$, Choquet integral of function f with respect to fuzzy measure μ is defined as follows, let $f\left(x_{(0)}\right) = 0$:

$$\int f d\mu = \sum_{j=1}^{k} \left[\left[f\left(x_{(j)}\right) - f\left(x_{(j-1)}\right)\right] \times \mu\left(A_{(j)}\right)\right], \tag{3}$$

where $\left\{x_{(1)}, x_{(2)}, \cdots, x_{(k)}\right\}$ is the reordered set of $\{x_1, x_2, \cdots, x_k\}$ as $f\left(x_{(1)}\right) \leq f\left(x_{(2)}\right) \leq \cdots \leq f\left(x_{(k)}\right)$, $A_{(j)} = \left\{x_{(j)}, x_{(j+1)}, \cdots, x_{(k)}\right\}$.

From Eq. (3) we can find that the determination of fuzzy measure μ is the key of calculation of Choquet integral. $f\left(x_{(j)}\right)$ is the known value which can be obtained according to the specially practical application.

III. Kernel Regression Ensemble Based on Fuzzy Integral-KREFI

A. Base-Regressors

In this paper, we use Priestley-Chao kernel estimator-PCKE [17] as the base-regressor. There are two commonly-used PCKEs which can be found in practical application, i.e., PCKE1 and PCKE2. Given the training dataset $\mathrm{D} = \{(x_i, y_i) | x_i \in \Re, y_i \in \Re, i = 1, 2, \cdots, n\}$, the regression functions obtained by PCKE1 and PCKE2 can be formulated as Eqs. (4) and (5) respectively (let $x_1 \leq x_2 \leq \cdots \leq x_n$):

$$m_{\mathrm{PCKE1}}(x) = \frac{\sum_{i=2}^{n} \left[(x_i - x_{i-1}) \mathrm{K}\left(\frac{x - x_i}{h}\right) y_i\right]}{h}, \tag{4}$$

and

$$m_{\mathrm{PCKE2}}(x) = \frac{\sum_{i=2}^{n-1} \left[(x_{i+1} - x_{i-1}) \mathrm{K}\left(\frac{x - x_i}{h}\right) y_i\right]}{2h}, \tag{5}$$

where $\mathrm{K}(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right)$ is Gaussian kernel and h is the bandwidth which is an important parameter impacting the performance of PCKEs. We use the cross-validation method to determine the optimal bandwidth $\hat{h}$ in Eqs. (4) and (5) as follows:

$$\hat{h}_{\mathrm{CV}} = \arg\min_{h \in \mathrm{H}} \{\mathrm{CV}(h)\}, \tag{6}$$

$$\mathrm{CV}(h) = \sum_{i=1}^{n} [y_i - m_{-i}(x_i)]^2, \tag{7}$$

where $m_{-i}(x)$ is the estimated regression function based on the dataset $\mathrm{D} - x_i$. Sometimes, the penalty function will be introduced to relax the roughness of bandwidth selected with Eq. (7):

$$\mathrm{CV}'(h) = \sum_{i=1}^{n} \left\{[y_i - m_{-i}(x_i)]^2 \pi(\mathrm{W}(x_i))\right\}, \tag{8}$$

Algorithm 1 Standard PSO-SPSO

1: **for** every particle x_i in swarm S **do**
2: $v_i = v_{\min} + (v_{\max} - v_{\min}) \cdot \mathrm{U}_i(0,1)$;
3: $x_i = x_{\min} + (x_{\max} - x_{\min}) \cdot \mathrm{U}_i(0,1)$;
4: $pb_i = x_i$; % Locally optimal solutions
5: **end for**
6: $pg = \arg\min_{x_i \in S} [f(x_i)]$; % Globally optimal solutions
7: **while** the termination condition does not hold **do**
8: **for** every particle x_i in swarm S **do**
9: $v_i = w \cdot v_i + c_1 \cdot u_1 \cdot (pb_i - x_i) + c_2 \cdot u_2 \cdot (pg - x_i)$;
10: $x_i = x_i + v_i$;
11: **if** $f(x_i) < f(pb_i)$ **then**
12: $pb_i = x_i$;
13: **if** $f(pb_i) < f(pg)$ **then**
14: $pg = pb_i$;
15: **end if**
16: **end if**
17: **end for**
18: **end while**
19: Output pg and $f(pg)$;

Algorithm 2 Gaussian PSO-GPSO

1: **for** every particle x_i in swarm S **do**
2: $v_i = v_{\min} + (v_{\max} - v_{\min}) \cdot \mathrm{U}_i(0,1)$;
3: $x_i = x_{\min} + (x_{\max} - x_{\min}) \cdot \mathrm{U}_i(0,1)$;
4: $pb_i = x_i$; % Locally optimal solutions
5: **end for**
6: $pg = \arg\min_{x_i \in S} [f(x_i)]$; % Globally optimal solutions
7: **while** the termination condition does not hold **do**
8: **for** every particle x_i in swarm S **do**
9: $v_i = |n_1| \cdot (pb_i - x_i) + |n_2| \cdot (pg - x_i)$;
10: $x_i = x_i + v_i$;
11: **if** $f(x_i) < f(pb_i)$ **then**
12: $pb_i = x_i$;
13: **if** $f(pb_i) < f(pg)$ **then**
14: $pg = pb_i$;
15: **end if**
16: **end if**
17: **end for**
18: **end while**
19: Output pg and $f(pg)$;

where $\pi(u) = (1-u)^{-2}$ is the penalty function and $\mathrm{W}(x) = \frac{\mathrm{K}(0)}{\sum_{i=1}^{n} \mathrm{K}\left(\frac{x-x_i}{h}\right)}$. This kind of cross-validation is also called generalized cross-validation.

We denote PCKE1 and PCKE2 with the optimal bandwidths determined by Eqs. (7) and (8) as $\mathrm{PCKE1_{CV}}$, $\mathrm{PCKE1_{GCV}}$, $\mathrm{PCKE2_{CV}}$ and $\mathrm{PCKE2_{GCV}}$ respectively. These four kernel regression estimators are served as the base-regressors of KREFI.

B. Description of KREFI Model

Assume the regression values estimated by $\mathrm{PCKE1_{CV}}$ (r_1), $\mathrm{PCKE1_{GCV}}$ (r_2), $\mathrm{PCKE2_{CV}}$ (r_3) and $\mathrm{PCKE2_{GCV}}$ (r_4) for a new sample x are $m(r_1)$, $m(r_2)$, $m(r_3)$ and $m(r_4)$ respectively. KREFI estimates its regression value y by integrating $m(r_1)$, $m(r_2)$, $m(r_3)$ and $m(r_4)$ with Choquet integral as the following Eq. (9), let $m\left(r_{(0)}\right) = 0$:

$$y = \int m d\mu = \sum_{j=1}^{4} \left[\left[m\left(r_{(j)}\right) - m\left(r_{(j-1)}\right) \right] \times \mu\left(R_{(j)}\right) \right], \tag{9}$$

where $\left\{r_{(1)}, r_{(2)}, r_{(3)}, r_{(4)}\right\}$ is the reordered set of $\{r_1, r_2, r_3, r_4\}$ as $m\left(r_{(1)}\right) \le m\left(r_{(2)}\right) \le m\left(r_{(3)}\right) \le m\left(r_{(4)}\right)$, $R_{(j)} = \left\{r_{(j)}, r_{(j+1)}, \cdots, r_{(4)}\right\}$.

From Eq. (9), we can find that the determination of fuzzy measure μ is very important to the estimation of regression value of new sample x. Based on information table as shown in TABLE I, we give the detailed explanation about how to determine the fuzzy measure for KREFI. In TABLE I, r_{ij} ($i = 1, 2, \cdots, n; j = 1, 2, 3, 4$) denotes the regression value of x_i estimated by regressor r_j based on the dataset $\mathrm{D} - x_i$. In our designed KREFI, there are 14 ($2^4 - 4 = 14$) measure values needed to be determined, i.e., $\mu(\{r_1\})$, $\mu(\{r_2\})$, $\mu(\{r_3\})$, $\mu(\{r_4\})$, $\mu(\{r_1, r_2\})$, $\mu(\{r_1, r_3\})$, $\mu(\{r_1, r_4\})$, $\mu(\{r_2, r_3\})$, $\mu(\{r_2, r_4\})$, $\mu(\{r_3, r_4\})$, $\mu(\{r_1, r_2, r_3\})$, $\mu(\{r_1, r_2, r_4\})$, $\mu(\{r_1, r_3, r_4\})$ and $\mu(\{r_2, r_3, r_4\})$. The optimal fuzzy measure can be found via the following optimization expression:

$$\min_{\mu} e = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - e_i)^2}, \tag{10}$$

where e_i including the unknown fuzzy measure can be calculated according to Eq. (3) or Eq. (9). Hence, the regression ensemble based on fuzzy integral is transformed to an optimization problem.

C. Particle Swarm Optimization-PSO

PSO [14] is designed by imitating the behavior of bird flocking, which use one globally optimal solution (*gbest*) and m locally optimal solutions (*pbest*) to guide the m birds (*particle*) to find the food (*optimal solution*) by adjusting their *velocities* and *positions* iteratively. In this paper, we will use three different PSO algorithms to optimize the expression Eq. (10), i.e., standard PSO-SPSO [14], Gaussian PSO-GPSO [15] and GPSO with Gaussian jump-GPSOGJ [16]. The corresponding algorithm procedures are summarized in Algorithms 1-3.

Now, we give some explanations to the parameters in these three algorithms. $v_{\max}$ and $v_{\min}$ are the upper and lower bounds of particle x_i's velocity respectively. And, $x_{\max}$ and $x_{\min}$ are the upper and lower bounds of particle x_i's position respectively. pb_i is the locally optimal solution of particle x_i and pg is the globally optimal solution of swarm S. In the line 9 of Algorithm 1, w is the inertia weight, c_1 and c_2 are acceleration constants, and u_1 and u_2 are the random numbers obeying the uniform distribution $\mathrm{U}(0,1)$. In the line 9 of Algorithm 2, n_1 and n_2 are the random numbers obeying the standard normal uniform distribution $\mathrm{N}(0,1)$. In the line 9 of Algorithm 3, $wait_i$ records the number of particle x_i falling into the local minimum and $wait_{\max}$ the maximal number of particle x_i falling into the local minimum. In the line 13 of Algorithm 3, $\eta \in [0,1]$ denotes the scale parameter.

TABLE I. INFORMATION TABLE OF KREFI

Training Sample	$\mathrm{PCKE1_{CV}}$ (r_1)	$\mathrm{PCKE1_{GCV}}$ (r_2)	$\mathrm{PCKE2_{CV}}$ (r_3)	$\mathrm{PCKE2_{GCV}}$ (r_4)	Integration value
x_1	r_{11}	r_{12}	r_{13}	r_{14}	e_1
x_2	r_{21}	r_{22}	r_{23}	r_{24}	e_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_n	r_{n1}	r_{n2}	r_{n3}	r_{n4}	e_n

TABLE II. 6 TESTING FUNCTIONS

Functions	Variable	Disturbance
$f_1 = 1 - x + \exp\left[-200\left(x-0.5\right)^2\right] + \varepsilon$	$x \sim \mathrm{U}\left(0,1\right)$	$\varepsilon \sim \mathrm{N}\left(0,0.1\right)$
$f_2 = x + \frac{\exp\left(-2x^2\right)}{\sqrt{2\pi}} + 0.2\varepsilon$	$x \sim \mathrm{U}\left(-0.5,2\right)$	$\varepsilon \sim \mathrm{N}\left(0,1\right)$
$f_3 = x + \sin\left[2\pi\left(1-x\right)^2\right] + 0.2\varepsilon$	$x \sim \mathrm{U}\left(-1,1\right)$	$\varepsilon \sim \mathrm{N}\left(0,1\right)$
$f_4 = x + 2\sin\left(15x\right) + 0.2\varepsilon$	$x \sim \mathrm{U}\left(-1,1\right)$	$\varepsilon \sim \mathrm{N}\left(0,1\right)$
$f_5 = \left[\sin\left(2\pi x^3\right)\right]^3 + 0.2\varepsilon$	$x \sim \mathrm{U}\left(-1,1\right)$	$\varepsilon \sim \mathrm{N}\left(0,1\right)$
$f_6 = \sin\left(3\pi x^2\right) + 0.2\varepsilon$	$x \sim \mathrm{U}\left(-1,1\right)$	$\varepsilon \sim \mathrm{N}\left(0,1\right)$

Algorithm 3 GPSO with Gaussian jump-GPSOGJ

for every particle x_i in swarm S **do**
  $v_i = v_{\min} + \left(v_{\max} - v_{\min}\right) \cdot \mathrm{U}_i\left(0,1\right)$;
  $x_i = x_{\min} + \left(x_{\max} - x_{\min}\right) \cdot \mathrm{U}_i\left(0,1\right)$;
  $pb_i = x_i$; % Locally optimal solutions
end for
$pg = \arg\min_{x_i \in S}\left[f\left(x_i\right)\right]$; % Globally optimal solutions
while the termination condition does not hold **do**
  for every particle x_i in swarm S **do**
    if $wait_i \le wait_{\max}$ **then**
      $v_i = \left|n_1\right| \cdot \left(pb_i - x_i\right) + \left|n_2\right| \cdot \left(pg - x_i\right)$;
      $x_i = x_i + v_i$;
    else
      $x_i = x_i + \eta \cdot \mathrm{N}_i\left(0,1\right)$; % Gaussian jump
    end if
    if $f\left(x_i\right) < f\left(pb_i\right)$ **then**
      $pb_i = x_i$;
      $wait_i = 0$;
      if $f\left(pb_i\right) < f\left(pg\right)$ **then**
        $pg = pb_i$;
      end if
    else
      $wait_i = wait_i + 1$;
    end if
  end for
end while
Output pg and $f\left(pg\right)$;

IV. SIMULATION RESULTS

To show the effectiveness and efficiency of the proposed KREFI models, the following simulations are performed.

A. Data Preparation

In our experiment, 6 testing functions listed in TABLE II are used to compare the performances of different regressors. For every testing function, 100 random samples are generated as training dataset which is used to learn the fuzzy measures and other 100 random samples are generated as testing dataset which is used to evaluate the mean squared errors (MSE) of different regressors. The finally experimental results are the average of 10-time evaluations.

B. Parameter Setups

The parameters in PSO algorithms are set up as follows:

1) The number of particles in swarm S is 50. And, the number of iterations is 30, i.e., $Iteration\_Num = 30$.
2) Because the fuzzy measure belongs to the interval $[0,1]$, $x_{\max} = 1$ and $x_{\min} = 0$. During the iteration, if $x_i > 1$ or $x_i < 0$, then we will let $x_i \leftarrow \frac{1}{1+e^{-x_i}}$ so that the position value which is larger than 1 or smaller than 0 can be reset into the interval $[0,1]$.
3) $v_{\max} = 0.5$ and $v_{\min} = -0.5$. If the updated velocity $v_i > v_{\max}$, then let $v_i = v_{\max}$; if the updated velocity $v_i < v_{\min}$, then let $v_i = v_{\min}$;
4) The inertia weight w gradually decreases from 0.9 to 0.4 with the increase of number of iterations, i.e., for the l-th iteration, $w \leftarrow 0.9 - \frac{0.9-0.4}{Iteration\_Num} \times l$.
5) Let the acceleration constants $c_1 = 2$ and $c_2 = 2$.
6) Let the scale parameter $\eta = 0.1$ and $wait_{\max} = 3$.

For the kernel regression ensemble based on general fuzzy measure $\mathrm{KREFI_g}$, the dimension of every particle is 14. And, for the kernel regression ensemble based on λ fuzzy measure KREFI_λ, the dimension of every particle is 4. It is worthwhile to note that the monotonicity of general fuzzy measure (i.e., the condition 2 of Definition 1) should be satisfied in the process of optimization for $\mathrm{KREFI_g}$. In order to implement this, we adopt the strategy used in work [22]. An array M is used to store the 14 fuzzy measure values. Any integer between 1 and 14 is transformed into a 4-bit binary number, e.g., $2 = 0\times2^0 + 1\times2^1 + 0\times2^2 + 0\times2^3$, $6 = 0\times2^0 + 1\times2^1 + 1\times2^2 + 0\times2^3$ and $11 = 1\times2^0 + 1\times2^1 + 0\times2^2 + 1\times2^3$. Then, $M\left[2\right] = \mu\left(\left\{r_2\right\}\right)$, $M\left[6\right] = \mu\left(\left\{r_2, r_3\right\}\right)$ and $M\left[11\right] = \mu\left(\left\{r_1, r_2, r_4\right\}\right)$. For $\forall i =$

$2, 3, \cdots, 14$ and $j = 1, 2, \cdots, i-1$, check whether $M[j] > M[i]$: if yes, exchange the values $M[j]$ and $M[i]$. We call this process "measure reordering" [22] which is conducted in every iteration. The optimization of KREFI_λ does not need the measure reordering, because there are only 4 measure values on singleton sets which are needed to be determined and other measures can be directly calculated according to Eqs. (1) and (2).

C. Experimental Results

Our experiment compares the performances (measured with MSE) of 11 different regression estimators, i.e., PCKE1_{CV}, $\text{PCKE1}_{\text{GCV}}$, PCKE2_{CV}, $\text{PCKE2}_{\text{GCV}}$, basic ensemble method (BEM) [7], KREFI_gs and KREFI_λs based on SPSO, GPSO and GPSOGJ, where BEM calculates the average of predictions of base-regressors in ensemble:

$$y = \frac{m(r_1) + m(r_2) + m(r_3) + m(r_4)}{4}, \qquad (11)$$

where the meanings of $m(r_1)$, $m(r_2)$, $m(r_3)$ and $m(r_4)$ have been given in Subsection III-B. In fact, we can find that there are strong dependence between these four base-regressors, i.e., PCKE1_{CV}, $\text{PCKE1}_{\text{GCV}}$, PCKE2_{CV}, $\text{PCKE2}_{\text{GCV}}$, because they are homologous, the main difference among them is the bandwidth selection. In addition, PCKE1_{CV}, $\text{PCKE1}_{\text{GCV}}$, PCKE2_{CV}, $\text{PCKE2}_{\text{GCV}}$ are high-variance, because PCKE is sensitive to the training data.

We firstly check the convergence of PSO algorithms in KREFI_gs and KREFI_λs based on these 6 testing functions. The experimental results are listed in Fig. 1. From these pictures, we can see that with the increase of iteration, the estimated errors in Eq. (10) corresponding to different PSO algorithms all gradually decrease first and then keep approximately smooth, which reflects that the fuzzy measures are updated gradually until the approximately optimal ones are found. This indicates that our proposed KREFI based on PSO algorithms are feasible. Then, we compare the regression performances of 11 regressors based on 6 testing functions. From the comparative results summarized in TABLE III, we can find that

1) KREFI_gs and KREFI_λs obtain the lower MSEs and variances compared with four base-regressors only with 30-time iterations. For example, MSE of KREFI_g + SPSO on f_1 is 0.030, which is lower than PCKE1_{CV}'s 0.050, $\text{PCKE1}_{\text{GCV}}$'s 0.064, PCKE2_{CV}'s 0.044, $\text{PCKE2}_{\text{GCV}}$'s 0.051 and BEM's 0.037; variance of KREFI_λ + SPSO on f_3 is 0.023, which is lower than PCKE1_{CV}'s 0.107, $\text{PCKE1}_{\text{GCV}}$'s 1.103, PCKE2_{CV}'s 0.044, $\text{PCKE2}_{\text{GCV}}$'s 0.071 and BEM's 0.155. This shows KREFI_gs and KREFI_λs are more accurate and stable than PCKEs and indicates our ensembles are effective and efficient. The main reason KREFI_gs and KREFI_λs can obtain the better regression performances is due to the ensemble which reduces estimation variances of individual regressors.
2) Compared with BEM which is the traditionally linear combination of base-regressors for regression ensemble, KREFI_gs and KREFI_λs also obtain the better regression performances. This shows KREFI_gs and KREFI_λs can effectively consider the dependence (or interaction) among base-regressors, because the weighted average in BEM can only consider the additive dependence, while fuzzy integral in KREFI_gs and KREFI_λs consider the non-additive dependence.
3) There is no obvious difference among the performances of different KREFI_gs and KREFI_λs. This conclusion also gives us an indication regarding the application of KREFI from another perspective: for the less number of iterations, SPSO, GPSO and GPSOGJ cannot remarkably affect the performances of developed KREFIs. For reference, we give the general and λ fuzzy measures in TABLE IV determined by SPSO, GPSO and GPSOGJ algorithms on f_3. From this table we can see that there are obvious differences among the fuzzy measures solved with SPSO, GPSO and GPSOGJ algorithms. However, the differences between some fuzzy measure values solved with the same PSO algorithm are small. This indicates that there are more than one optimal fuzzy measure for KREFI and the fuzzy measure is sensitive to the initialization of PSO algorithm.

V. Conclusions

In this paper, a new kernel regression ensemble model based on fuzzy integral (KREFI) is proposed, which considers the dependence or interaction among the different base-regressors so that the importance of each individual regressor can be expressed clearly. The fuzzy measure is the key for the application of Choquet fuzzy integral. In order to determine two different kinds of fuzzy measures (i.e., general and λ fuzzy measures) in KREFI, three PSO algorithms are adopted. The finally experimental results reveal that KREFI models are more accurate and stable in comparison with the base-regressors and commonly-used basic regression method. A number of enhancements and future research can be summarized as follows: (1) using other kinds of fuzzy integrals, e.g., Sugeno [23] and upper [24] integrals, to construct the regression ensemble models; (2) designing a fuzzy integral based measure to evaluate the interaction among base-regressors and study how this interaction affects the regression ensemble and (3) seeking the practical application for KREFI, e.g., the regression analysis to Fourier transform infrared spectroscopy in the field of optical engineering.

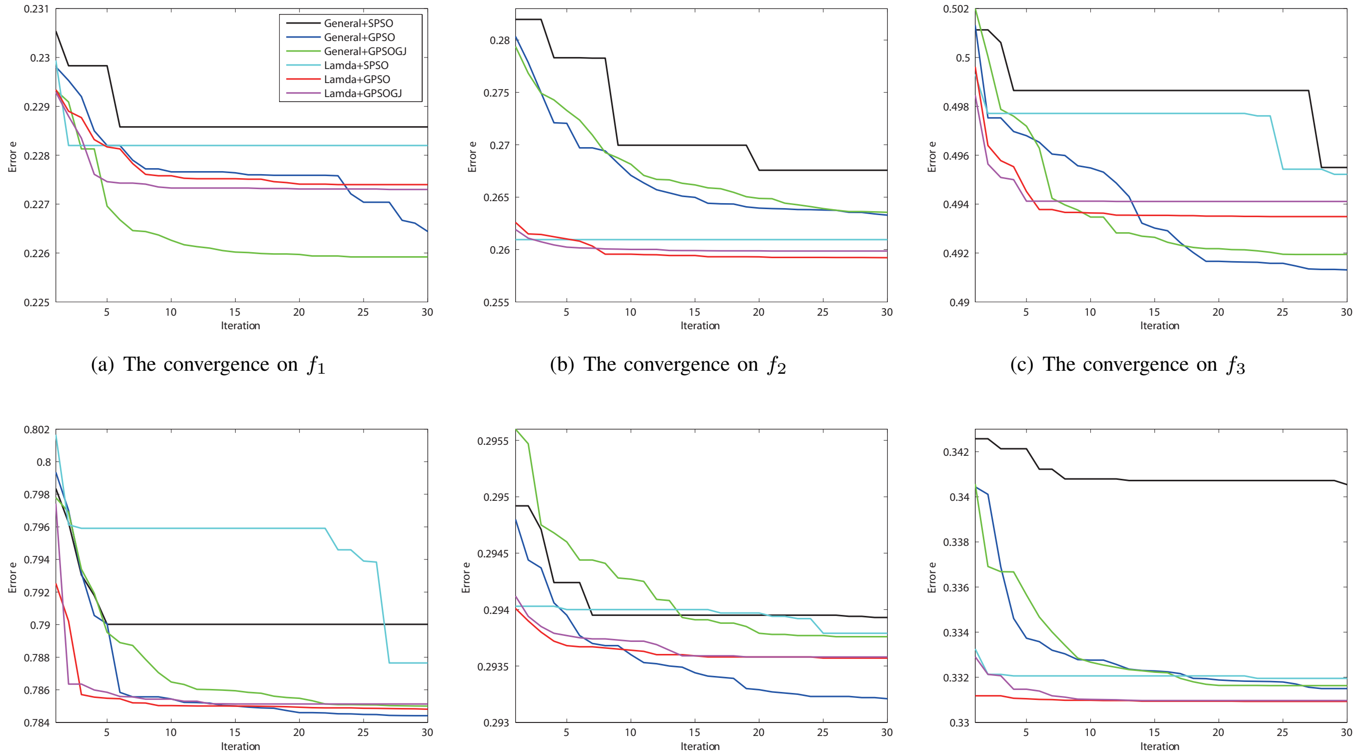
Acknowledgment

The authors thank three anonymous reviewers. Their valuable and constructive comments and suggestions helped us in significantly improving the manuscript. This work was supported in part by the CRG grants G-YL14 and G-YM07 of The Hong Kong Polytechnic University and by the National Natural Science Foundations of China under Grant 61170040 and Grant 71371063.

References

[1] Z.H. Zhou, *Ensemble Methods: Foundations and Algorithms.* Chapman & Hall, Boca Raton, USA, 2012.

[2] C. Zhang and Y.Q. Ma, *Ensemble Machine Learning: Methods and Applications.* Springer, New York, USA, 2012.



(a) The convergence on f_1

(b) The convergence on f_2

(c) The convergence on f_3

(d) The convergence on f_4

(e) The convergence on f_5

(f) The convergence on f_6

Fig. 1. The convergence of PSO algorithms based KREFI models on 6 testing functions

[3] G. Brown, J.L. Wyatt and P. Tiňo, “Managing diversity in regression ensembles,” *The Journal of Machine Learning Research,* vol. 6, pp. 1621-1650, 2005.

[4] R.E. Banfield, L.O. Hall, K.W. Bowyer and W.P. Kegelmeyer, “A comparison of decision tree ensemble creation techniques,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 29, no. 1, pp. 173-180, 2007.

[5] Z.H. Zhou, J.X. Wu and W. Tang, “Ensembling neural networks: Many could be better than all,” *Artificial Intelligence,* vol. 137, no. 1-2, pp. 239-262, 2002.

[6] H.C. Kim, S.N. Pang, H.M. Je, H.M. Kim and S.Y. Bang, “Constructing support vector machine ensemble,” *Pattern Recognition,* vol. 36, no. 12, pp. 2757-2767, 2003.

[7] J.M. Moreiraa, C. Soaresb, A.M. Jorgeb and J.F. de Sousaa, “Ensemble approaches for regression: a survey,” *ACM Computing Surveys*, vol. 45, no. 1, article 10, 2012.

[8] M. Grabisch, “Fuzzy integral in multicriteria decision making,” *Fuzzy Sets and Systems,* vol. 69, no. 3, pp. 279-298, 1995.

[9] S.B. Cho and J.H. Kim, “Combining multiple neural networks by fuzzy integral for robust classification,” *IEEE Transactions on Systems, Man and Cybernetics,* vol. 25, no. 2, pp. 380-384, 1995.

[10] M. Grabisch, “A new algorithm for identifying fuzzy measures and its application to pattern recognition,” *In Proceedings of FUZZ-IEEE/IFES’95,* vol. 1, pp. 145-150, 1995.

[11] R. Yang, Z. Wang, P.A. Heng and K.S. Leung, “Fuzzified Choquet integral with a fuzzy-valued integrand and its application on temperature prediction,” *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics,* vol. 38, no. 2, pp. 367-380, 2008.

[12] J. Wang and Z. Wang, “Using neural networks to determine Sugeno measures by statistics,” *Neural Networks,* vol. 10, no. 1, pp. 183-197, 1997.

[13] X.Z. Wang, Y.L. He, L.C. Dong and H.Y. Zhao, “Particle swarm optimization for determining fuzzy measures from data,” *Information Sciences,* vol. 181, no. 19, pp. 4230-4252, 2011.

[14] J. Kennedy and R.C. Eberhart, “Particle swarm optimization,” *In Proceedings of 1995 IEEE International Conference on Neural Networks,* vol. 4, pp. 1942C1948, 1995.

[15] R.A. Krohling, “Gaussian swarm a novel particle swarm optimization algorithm,” *In Proceedings of 2004 IEEE Conference on Cybernetics and Intelligent Systems,* vol. 1, pp. 372-376, 2004.

[16] R.A. Krohling, “Gaussian particle swarm with jumps,” *In Proceedings of 2005 IEEE Congress on Evolutionary Computation,* vol. 2, pp. 1226-1231, 2005.

[17] M.B. Priestley and M.T. Chao, “Non-parametric function fitting,” *Journal of the Royal Statistical Society, Series B (Methodological),* vol. 34, no. 3, pp. 385-392, 1972.

[18] L.J. Wang, “An improved multiple fuzzy NNC system based on mutual information and fuzzy integral,” *International Journal of Machine Learning and Cybernetics,* vol. 2, no. 1, pp. 25-36, 2011.

[19] C.M. He, “Approximation of polygonal fuzzy neural networks in sense of Choquet integral norms,” *International Journal of Machine Learning and Cybernetics,* DOI: 10.1007/s13042-013-0154-8, 2013.

[20] Z.Y. Wang and G.J. Klir, *Fuzzy measure theory.* Plenum Press (A division of Plenum Publishing Corporation), New York, 1992.

[21] V. Torra and Y. Narukawa, *Modeling decisions: information fusion and aggregation operators (Cognitive Technologies).* Springer, 2007.

[22] Z.Y. Wang, K.S. Leung, M.L. Wong, J. Fang and K.B. Xu, “Nonlinear nonnegative multiregressions based on Choquet integrals,” *International Journal of Approximate Reasoning,* vol. 25, no. 2, pp. 71-87, 2000.

[23] J.L. Marichal, “On Choquet and Sugeno integrals as aggregation functions,” *Fuzzy Measures and Integrals*, vol. 40, pp. 247-272, 2000.

[24] Z.Y. Wang, W.Y. Li, K.H. Lee and K.S. Leung, “Lower integrals and upper integrals with respect to nonadditive set functions,” *Fuzzy Sets and Systems*, vol. 159, no. 6, pp. 646-660, 2008.

TABLE III. MSEs OF DIFFERENT REGRESSION ESTIMATORS ON 6 STANDARD TESTING FUNCTIONS

	$PCKE1_{CV}$	$PCKE1_{GCV}$	$PCKE2_{CV}$	$PCKE2_{GCV}$	BEM	$KREFI_g$ + SPSO	$KREFI_g$ + GPSO	$KREFI_g$ + GPSOGJ	$KREFI_\lambda$ + SPSO	$KREFI_\lambda$ + GPSO	$KREFI_\lambda$ + GPSOGJ
f_1	0.050±0.025	0.064±0.028	0.044±0.022	0.051±0.024	0.037±0.016	0.030±0.007	0.029±0.010	0.031±0.010	0.030±0.011	0.033±0.012	0.034±0.011
f_2	0.111±0.083	0.234±0.182	0.145±0.100	0.149±0.086	0.131±0.082	0.074±0.028	0.093±0.069	0.081±0.036	0.129±0.074	0.144±0.119	0.111±0.082
f_3	0.263±0.107	1.039±1.103	0.191±0.044	0.274±0.071	0.260±0.155	0.172±0.041	0.173±0.025	0.171±0.023	0.167±0.023	0.176±0.039	0.177±0.039
f_4	0.728±0.303	0.803±0.362	0.713±0.449	0.988±0.517	0.677±0.313	0.544±0.400	0.575±0.454	0.569±0.445	0.481±0.353	0.593±0.457	0.596±0.467
f_5	0.081±0.025	0.114±0.060	0.078±0.029	0.095±0.031	0.087±0.045	0.075±0.023	0.074±0.023	0.074±0.024	0.075±0.024	0.077±0.024	0.076±0.024
f_6	0.103±0.019	0.136±0.029	0.101±0.017	0.111±0.030	0.109±0.031	0.093±0.030	0.089±0.022	0.088±0.023	0.087±0.018	0.093±0.023	0.092±0.023

TABLE IV. GENERAL AND λ FUZZY MEASURES DETERMINED BY SPSO, GPSO AND GPSOGJ ALGORITHMS ON f_3

	$\mu(\{r_1\})$	$\mu(\{r_2\})$	$\mu(\{r_1,r_2\})$	$\mu(\{r_3\})$	$\mu(\{r_1,r_3\})$	$\mu(\{r_2,r_3\})$	$\mu(\{r_1,r_2,r_3\})$	$\mu(\{r_4\})$	$\mu(\{r_1,r_4\})$	$\mu(\{r_2,r_4\})$	$\mu(\{r_1,r_2,r_4\})$	$\mu(\{r_3,r_4\})$	$\mu(\{r_1,r_3,r_4\})$	$\mu(\{r_2,r_3,r_4\})$
$KREFI_g$ + SPSO	0.00119	0.00120	0.00239	0.99436	0.99438	0.99438	0.99440	0.32715	0.32795	0.32796	0.32877	0.99997	0.99999	0.99999
$KREFI_g$ + GPSO	0.99517	0.19095	0.99610	0.98492	0.99997	0.98780	0.99998	0.30019	0.99663	0.43382	0.99728	0.98946	0.99999	0.99148
$KREFI_g$ + GPSOGJ	0.20850	0.00030	0.20880	0.80848	0.99879	0.80875	0.99905	0.00106	0.20954	0.00136	0.20983	0.80945	0.99973	0.80972
$KREFI_\lambda$ + SPSO	0.12666	0.00029	0.12708	0.58572	0.99602	0.58665	0.99741	0.00054	0.12745	0.00083	0.12788	0.58747	0.99861	0.58840
$KREFI_\lambda$ + GPSO	0.98728	0.67500	0.99656	0.76521	0.99780	0.92423	0.99999	0.01006	0.98742	0.67828	0.99661	0.76758	0.99784	0.92500
$KREFI_\lambda$ + GPSOGJ	0.30831	0.03100	0.32976	0.97381	0.98205	0.97464	0.98263	0.96970	0.97921	0.97066	0.97987	0.99973	0.99998	0.99975