Abstract—In evolutionary optimization surrogate models are commonly used when the evaluation of a fitness function is computationally expensive. Here the fitness of individuals are indirectly estimated by modeling their rank with respect to the current population by use of ordinal regression. This paper focuses on how to validate the goodness of fit for surrogate models during search and introduces a novel validation/updating policy for surrogate models, and is illustrated on classical numerical optimization functions for evolutionary computation. The study shows that for validation accuracy it is sufficient for the approximate ranking and true ranking of the training set to be sufficiently concordant or that only the potential parent individuals should be ranked consistently. Moreover, the new validation approach reduces the number of fitness evaluation needed, without a loss in performance.

Keywords—surrogate models; ordinal regression; sampling; evolutionary optimization

I. INTRODUCTION

Evolutionary optimization is a stochastic and direct search method where a population of individuals are searched in parallel. Typically only the full or partial ordering of these parallel search individuals is needed. For this reason an ordinal regression offers sufficiently detailed surrogates for evolutionary computation [1]. In this case there is no explicit fitness function defined, but rather an indirect method of evaluating whether one individual is preferable to another.

The current approach in fitness approximation for evolutionary computation involves building surrogate fitness models directly using regression. For a recent review of the state-of-the-art surrogate models see [2]–[5]. The fitness model is based on a set of evaluated solutions called the training set. The surrogate model is used to predict the fitness of candidate search individuals. Commonly a fraction of individuals are selected and evaluated within each generation (or over some number of generations [6]), added to the training set, and used for updating the surrogate. The goal is to reduce the number of costly true fitness evaluations while retaining a sufficiently accurate surrogate during evolution. When using ordinal regression a candidate search individual \(x_i\) is said to be preferred over \(x_j\) if \(x_i\) has a higher fitness than \(x_j\). The training set for the surrogate model is therefore composed of pairs of individuals \((x_i, x_j)\) and a corresponding label \(t_k \in [1, -1]\), taking the value +1 (or −1) when \(x_i\) has a higher fitness than \(x_j\) (or vice versa). The direct fitness approximation approach does not make full use of the flexibility inherent in the ordering requirement. The technique used here for ordinal regression is kernel based and is described in section II and was first presented in [1]. The use of surrogate models and approximate ranking has made some headway, e.g. [7], however still remains relatively unexplored field of study.

The critical issue in generating surrogate models, for evolutionary strategy (ES) search [8], is the manner in which the training set is constructed. For example, in optimization it is not critical to model accurately regions of the search space with low fitness. It is, however, key to model accurately new search regions deemed potentially lucrative by the evolutionary search method. Furthermore, since the search itself is stochastic, perhaps the ranking need not to be that accurate. Indeed the best \(\mu\) candidate individuals are commonly selected and the rest disregarded irrespective of their exact ranking.

In the literature new individuals are added to the training set from the new generation of unevaluated search individuals. This seems sensible since this is the population of individuals which need to be ranked. However, perhaps sampling a representative individual, for example the mean of the unevaluated search individuals, may also be useful in surrogate ranking. Typically, the unevaluated individuals are ranked using the current surrogate model and then the best of these are evaluated using the true expensive fitness function and added to the training set. Again, this seems sensible since we are not interesting in low fitness regions of the search space. Nevertheless, it remains unclear whether this is actually the case. Finally, there is the question of knowing when to stop, when is our surrogate sufficiently accurate? Is it necessary to add new search individuals to our training set at every search generation? What do we mean by sufficiently accurate? This paper describes some preliminary experiments with the aim of investigating some of these issues further.

In section III sampling methods, stopping criteria and model accuracy are discussed. Moreover, a strategy for updating the surrogate during search is presented and its effectiveness illustrated using CMA-ES on some numerical optimization functions in section IV. The paper concludes with discussion and summary in section V.

II. ORDINAL REGRESSION

Ordinal regression in evolutionary optimization has been previously presented in [1], but is given here for completeness. The ranking problem is specified by a set \(S = \{(x_i, y_i)\}_{i=1}^\nu \subset \mathbb{R}^d \times \{-1, 1\}\)
where the symbol \( \triangleright \) denotes “is preferred to”. In ordinal regression the task is to obtain function \( h \) that can for a given pair \((x_i, y_i)\) and \((x_j, y_j)\) distinguish between two different outcomes: \( y_i > y_j \) and \( y_j > y_i \). The task is, therefore, transformed into the problem of predicting the relative ordering of all possible pairs of adjacent ranks, \( (y_i > y_j, y_j > y_i) \). Note that \( y_i \) is the number of features. Then the surrogate considered may be defined by a linear function in the kernel-defined feature space:

$$ h(x) = \sum_{i=1}^{m} w_i \phi_i(x) = \langle w \cdot \phi(x) \rangle. $$

where \( w = [w_1, ..., w_m] \in \mathbb{R}^m \) has weight \( w_i \) corresponding to feature \( \phi_i \).

The aim now is to find a function \( h \) that encounters as few training errors as possible on \( S' \). Applying the method of large margin rank boundaries of ordinal regression described in [9], the optimal \( w^* \) is determined by solving the following task:

$$ \min_w \frac{1}{2} \langle w \cdot w \rangle + \frac{\xi}{2} \sum_{k=1}^{\ell'} \xi_k^2 $$

subject to \( t_k \langle w \cdot (\phi(x_k^{(1)}) - \phi(x_k^{(2)})) \rangle \geq 1 - \xi_k \) and \( \xi_k \geq 0 \), \( k = 1, \ldots, \ell' \). The degree of constraint violation is given by the margin slack variable \( \xi_k \) and when greater than 1 the corresponding pair are incorrectly ranked. Note that \( h(x_i) - h(x_j) = \langle w \cdot (\phi(x_i) - \phi(x_j)) \rangle \)

and that minimizing \( \langle w \cdot w \rangle \) maximizes the margin between rank boundaries, in our case the distance between adjacent ranks \( h(x_k^{(1)}) \) and \( h(x_k^{(2)}) \).

Furthermore, it is important to scale the features \( \phi \) first as the evolutionary search zooms in on a particular region of the search space. A standard method of doing so is by scaling the training set such that all solutions are in some range, typically \([-1, 1]\). That is, scaled \( \phi \) is

$$ \hat{\phi}_i = 2(\phi_i - \underline{\phi})/(\bar{\phi} - \underline{\phi}) - 1 \quad i = 1, \ldots, m $$

where \( \underline{\phi}, \bar{\phi} \) are the minimum and maximum \( i \)-th component of all feature vectors in the training set.
the training set should be replaced, and whether the outgoing training individuals should be the worst ranking ones (elitist) or chosen at random (universal), where the elitist perspective was considered more favorable. However, reevaluating a subset of the best ranked individuals w.r.t. the surrogate model with the exact fitness function yielded the greatest performance edge of the strategies explored.

When the training accuracy is 100% one way of evaluating the accuracy of the surrogate is through cross validation. The quality of the surrogate is measured as the rank correlation between the surrogate ranking and the true ranking on training data. Here Kendall’s τ is used for this purpose [16]. Kendall’s τ is computed using the relative ordering of the ranks of all \( \ell(\ell - 1)/2 \) possible pairs. A pair is said to be concordant if the relative ranks of \( h(x_i) \) and \( h(x_j) \) are the same for \( f(x_i) \) and \( f(x_j) \), otherwise they are discordant. Kendall’s τ is the normalized difference in the number of concordant and discordant pairs, defined as follows,

\[
\tau = \frac{C - D}{\sqrt{C + D + T(h)} \sqrt{C + D + T(f)}}
\]

where \( C \) and \( D \) denote the number of concordant and discordant pairs, respectively, and \( T \) denotes the number of ties. Two rankings are the same when \( \tau = 1 \), completely reversed if \( \tau = -1 \), and uncorrelated for \( \tau \approx 0 \).

The surrogate ranking validation and improvement strategy using ordinal regression is tested using a covariance matrix adaptation evolution strategy (CMA-ES) [17]. CMA-ES is a very efficient numerical optimization technique, however we still expect to reduce the number of function evaluations needed for search. In [1] the validation policy had to successfully rank all of the candidate individuals, i.e. until \( \tau = 1 \). If there is no limit to training size then updating the surrogate becomes too computationally expensive, hence the training size needs to be pruned to size to \( \ell \). In [1] the set was pruned to a size \( \ell = \lambda \) by omitting the oldest individuals first. These are quite stringent restrictions which can be improved upon. The pruning only considers the age of the individuals, however older individuals might still be of more interest than newer ones if their fitness ranks higher. Thus a more sophisticated way of pruning would be omitting the lowest ranking individuals first. Moreover, candidate individuals are generated randomly using a normal distribution, thus a pseudo individual representing their mean could be of interest as an indicator for the entire population, e.g. by validating this pseudo individual first could give information if the surrogate is outdated w.r.t. the current search space. Furthermore, the validation is only done on the candidate individuals for the current generation in ES where only the \( \mu \) best ranked individuals will survive to become parents. In evolutionary computing one is interested in the accurate ranking of individuals generated in the neighborhood of parent individuals, hence for sufficient validation of the surrogate, only the \( \mu \) best ranked individuals should be considered and evaluated, since all other individuals of lower rank will be disregarded in the next iteration of ES. Lastly, one should also investigate the frequency by which the model is validated, e.g. at each generation or every \( K \) generations or even have the need for validating adapt with time.

Preliminary tests were conducted on which validation method deemed fruitful, by implementing Rosenbrock’s function of dimension \( n = 2 \), for 1) the setup presented in [1] and comparing it with the aforementioned validation improvements, which were added one at a time. Namely; 2) omitting the worst individuals during the pruning process, instead of the oldest ones; 3) initialize the validation process by using a pseudo individual that represents the mean of the new candidate individuals; 4) requiring that only the \( \mu \) best candidate individuals are correctly ranked; and 5) validating on every other generation. Experimental results focusing on the number of function evaluations are shown in Fig. 1. There is no statistical difference between omitting oldest or worst ranked individuals from the training set, but this was expected, since both are believed to be representatives of a region of the search space which is no longer of interest. Adding the pseudo mean candidate individual didn’t increase the performance edge. When the surrogate was updated on every other generation, it quickly became outdated and more than double function evaluations were needed to achieve the same rate of convergence. However, requiring the correct ranking for only the \( \mu \) best ranked candidate individuals showed a significant performance edge.

If the training accuracy is not 100% then clearly \( \tau < 1 \). In this case additional training individuals would be forced for evaluation. However, enforcing a completely concordant ranking, i.e. \( \tau = 1 \), was deemed to be too strict due to the fact the search is stochastic. Thus the surrogate is said to be sufficiently accurate if \( \tau > 0.999 \).

Based on these preliminary tests, a pseudo code for the proposed model validation and improvement strategy is described in Fig. 2 where it is implemented at the end of each generation of CMA-ES. The algorithm essentially only evaluates the expensive true fitness function when the surrogate is believed.
Initialization: Let \( Y \) denote current training set and its corresponding surrogate by \( h \). Let \( X \) denote population of \( \lambda \) individuals of unknown fitness under inspection.

\[
\begin{align*}
&\text{for } t := 1 \text{ to } \lambda \text{ do (validate a test individual)} \quad &\text{estimate ranking of } X \text{ using } h; \text{ denoted by } \tilde{R}_0. \\
&\quad x_B \leftarrow \max_{x \in X, y} \{ R_0 \} \text{ (test individual).} \\
&\quad \text{rank } x_B \text{ w.r.t. individuals in } Y \text{ using } h; \text{ denoted by } \tilde{R}. \\
&\quad \text{evaluate } x_B \text{ using true fitness function and evaluate its true rank among individuals in } Y; \text{ denoted by } R. \\
&\quad Y' \leftarrow Y \cup \{ x_B \} \text{ (add to training set).} \\
&\quad \text{compare the rankings } R \text{ and } \tilde{R} \text{ by computing the rank correlation } \tau. \\
&\quad \text{if } \tau > 0.999 \text{ then} \\
&\quad \quad \text{break (model is sufficiently accurate)} \\
&\quad \text{fi} \\
&\quad \text{update the surrogate } h \text{ using the new training set } Y. \\
&\quad \text{if } \mu \text{ best individuals of } \tilde{R}_0 \text{ have been evaluated then} \\
&\quad \quad \text{break (model is sufficiently accurate).} \\
&\quad \text{fi} \\
&\text{od}
\end{align*}
\]

Fig. 2. Sampling strategy to validate and improve surrogate models.

to have diverged. During each iteration of the validation process there are two sets of individuals, \( Y \) and \( X \), which are the training individuals which have been evaluated with the expensive model, and the candidate individuals (of unknown fitness) for the next iteration of CMA-ES, respectively. The test individuals of interest are those who are believed to become parent individuals in the next generation of CMA-ES, i.e. the \( \mu \) best ranked candidate individuals according to the surrogate \( h \). The method uses only a simple cross-validation on a single test individual, the one which the surrogate ranks the highest and has not yet been added to the training set. Creating more test individuals would be too costly, but plausible. Once a test individual has been evaluated it is added to the training set and the surrogate \( h \) is updated w.r.t. \( Y \), cf. Fig. 3. This is repeated until the surrogate is said to be sufficiently accurate, which occurs if either:

- Kendall’s \( \tau \) statistic between the ranking of the training set using the surrogate, \( \tilde{R} \), and its true ranking, \( R \), is higher than 0.999, or
- \( \mu \) best ranked candidate individuals w.r.t. the current surrogate have been added to the training set.

Note that during each update of the surrogate of the ranking of the \( \mu \) best candidate individuals can change. Thus it is possible to evaluate more then \( \mu \) test individuals during each validation.

Once the validation algorithm has completed, the training set is pruned to a size \( \bar{\ell} = \lambda \) by omitting the lowest ranking individuals .

IV. EXPERIMENTAL STUDY

In the experimental study CMA-ES is run for several test functions, namely sphere model and Rosenbrock’s function, of various dimensions \( n = 2, 5, 10 \) and 20. The average fitness for 100 independent runs versus the number of function evaluations is reported using the original validation procedure presented in [1] and compared with its new and improved validation procedure presented in Fig. 2, the procedures will be referred to as using “all” or only the “\( \mu \) best” candidate individuals during the validation, respectively. The parameter setting for the (\( \mu, \lambda \)) CMA-ES is as recommended in [17] with population size \( \lambda = 4 + [3 \ln(n)] \) and the number of parents selected \( \mu = \lambda/4 \). The stopping criteria used are 1000 function evaluation or a fitness less than \( 10^{-10} \). The initial mean search individual is generated from a uniform distribution between 0 and 1. It is also noted that the training set is only pruned to size \( \bar{\ell} = \lambda \) subsequent to the validation and improvement procedure introduced in Fig. 2.

A. Sphere model

The first experimental results are presented for the unimodal sphere model of dimension \( n \),

\[
f(x) = \sum_{i=1}^{n} x_i^2. \tag{8}
\]

The average fitness versus the number of function evaluations is presented in Fig. 4. A performance edge is achieved by restricting the validation strategy to only having the surrogate correctly rank the \( \mu \) highest ranking individuals, and thereby saving the algorithm of evaluating individuals that would have been disregarded in the next iteration. Fig. 5 shows the mean intermediate function evaluations that are calculated during the validation process. As one expects, requiring the method to evaluate no more than the \( \mu \) best ranked candidate individuals results in a lower intermediate function evaluations, generally saving the method one function evaluation per generation, it also achieves a better mean fitness, as shown in Table I.

B. Rosenbrock’s function

The first experiment is now repeated for Rosenbrock’s function,

\[
f(x) = \sum_{i=2}^{n} 100(x_i - x_{i-1}^2)^2 + (1 - x_{i-1})^2. \tag{9}
\]

The average fitness versus the number of function evaluations is presented in Fig. 6 and Fig. 7 shows the mean intermediate function evaluations that are calculated during the
validation process. Despite requiring more generations, the over all function evaluations are significantly lower and yield a better fitness when updating the surrogate on only the $\mu$ best individuals as shown in Table II. If all of the candidate individuals have to be ranked correctly, the method will get stuck in local minima for this problem in around 6 out of 100 experiments, however this is not a problem if only the $\mu$ best candidate individuals are ranked consistently, except at high dimensions, and even then the $\mu$ best individuals policy significantly outperforms evaluating all of the candidate individuals. Clearly the choice of validation policy will influence search performance.
instead of only focusing on consistently ranking the \( \mu \) best candidate individuals. Therefore, one can take inspiration from a varying random walk population model suggested by [19] to approximate the population sizing to overcome unnecessary fitness evaluations.

**References**


