Search Space Boundaries in Neural Network Error Landscape Analysis

Anna Sergeevna Bosman
Department of Computer Science
University of Pretoria
Pretoria, South Africa
Email: annar@cs.up.ac.za

Andries Engelbrecht
Department of Computer Science
University of Pretoria
Pretoria, South Africa
Email: engel@cs.up.ac.za

Mardé Helbig
Department of Computer Science
University of Pretoria
Pretoria, South Africa
Email: mhelbig@cs.up.ac.za

Abstract—**Fitness landscape analysis encompasses a selection of techniques designed to estimate the properties of a search landscape associated with an optimisation problem. Applied to neural network training, fitness landscape analysis can be used to establish the link between the shape of the objective function and various neural network design and architecture properties. However, most fitness landscape analysis metrics rely on search space sampling. Since neural network search space is unbounded, it is unclear what subset of the search space should be sampled to obtain representative measurements. This study analyses fitness landscape properties of neural networks under various search space boundaries, and proposes meaningful search space bounds for neural network fitness landscape analysis.**

I. Introduction

Neural networks (NNs) have been studied and successfully used in numerous practical applications for decades [1], [2], yet the landscape properties of the objective functions associated with supervised NN training are still poorly understood [3]. The inherent high dimensionality of NNs prevents intuitive visualisation, and NNs are often treated as “black box” optimisation problems as a result. The presence of saddle points [4], [5], as well as plateaus and narrow ridges [6], [7] in NN error surfaces has been established, but the relationship between these features and NN parameters such as the number of neurons, the number of hidden layers, activation function choice, etc., is still unclear.

One way to empirically study the link between the objective function landscape characteristics and the different NN parameters is through fitness landscape analysis (FLA). FLA is a young and evolving field of computational intelligence, first applied in evolutionary computation for algorithm performance prediction [8], [9]. The aim of FLA is to estimate and quantify topographical properties of the given objective function landscape in order to better understand the optimisation problem at hand [10], [11].

Fitness landscape properties are estimated by taking random samples of the search space, calculating the objective function value for every point in each sample, and analysing the relationship between the spatial and the qualitative characteristics of the sample points. In the context of NNs, the search space is made up of all possible weight combinations, and the NN error measures corresponding to these weight combinations make up the objective function landscape, also referred to as the error landscape.

An important difference between NN training and many other real-world continuous optimisation problems is that the NN search space is unbounded. Even though the weights are usually initialised on a small region around the origin [12], the weights may take on any values in the course of training. How can such a search space be sampled in a representative way? If random sampling techniques are used to estimate the FLA error landscape properties, what part of the infinite search space should be considered? This work attempts to answer the above questions by studying the FLA properties of a selection of NN error surfaces under different search space boundaries.

The rest of the paper is structured as follows: Section II describes FLA measures used in this paper. Section III discusses how FLA is applied to NN error surfaces. Section IV describes the experimental procedure followed. Section V presents the empirical study of the influence of search space boundaries on the FLA measurements. Section VI concludes the paper.

II. Fitness Landscape Analysis

A fitness landscape is a representation of the search space with regards to the objective function fitness values [8], [13]. The term was coined in the evolutionary optimisation community, but is applicable to any optimisation problem with a well-defined objective function. The objective function values calculated across the search space form a hypersurface that the optimisation algorithm either minimises or maximises. The aim of FLA is to estimate landscape features of the objective function landscape in order to gain insight into the structure of the problem and to better understand why a given algorithm performs well or fails [10], [11].

The rest of this section describes the FLA measures used in this study.

A. Gradients

An important characteristic of a fitness landscape is the speed with which the fitness changes as the landscape is traversed, i.e. the fitness gradient. Malan and Engelbrecht [14] proposed two gradient measures to quantify the magnitude of the fitness changes: the average estimated gradient, G_{avg}, and the standard deviation of the gradient, G_{dev}. G_{avg} and G_{dev}

are calculated based on Manhattan random walks [14] through the search space. Applied to NN error landscapes, G_{avg} is defined as:

$$G_{avg} = \frac{\sum_{t=0}^{T-1} |g(t)|}{T}$$

where T is the number of steps in the random walk, and $g(t)$ is defined as:

$$g(t) = \frac{\Delta e_t}{dist_t}$$

where Δe_t is the difference between the error values of the weight vectors defining step t of the random walk, and $dist_t$ is the Euclidean distance between the start point and the end point of step t. Similarly, G_{dev} is defined as:

$$G_{dev} = \sqrt{\frac{\sum_{t=0}^{T-1} (G_{avg} - |g(t)|)^2}{T-1}}$$

The G_{avg} measurement is essentially the mean magnitude of change in fitness values, while G_{dev} is the corresponding standard deviation. Lower values for G_{avg} and G_{dev} typically indicate a simpler landscape that is easier to search. However, a lack of gradients can also mean high neutrality (plateaus) in the landscape [6], which may prove challenging to optimisation algorithms that use gradient information to guide the search, such as gradient descent.

B. First entropic measure of ruggedness

Another important property of a fitness landscape is the frequency of change. A fitness landscape where the fitness monotonously decreases or increases is much easier to minimise or maximise than a fitness landscape where the fitness goes up and down all the time. The amount of change in the landscape can be referred to as the degree of ruggedness present in the landscape. Malan and Engelbrecht proposed two ruggedness measures based on Vassilev's [15] first entropic measure (FEM). These measures are based on a progressive random walk [16] through the search space, and quantify the change in fitness values based on entropy [17]. The two entropic measures of ruggedness used are:

- $FEM_{0.01}$ – Micro-ruggedness, based on a random walk with a maximum step size of 1% of the bounded search space.
- $FEM_{0.1}$ – Macro-ruggedness, based on a random walk with a maximum step size of 10% of the bounded search space.

The value of FEM is continuous and ranges between 0 and 1, where 0 indicates a flat landscape, and 1 indicates maximal ruggedness.

C. Fitness distance correlation

The fitness distance correlation (FDC) metric was proposed by Jones [18] as a measure of global problem hardness. FDC gives an indication of the global shape of the landscape to be searched. FDC measures the covariance between the fitness of a solution and its distance to the nearest optimum.

Malan and Engelbrecht [19] proposed a new measure, FDC$_s$, that is based on a sample of solutions without known optima. Applied to NN error landscapes, FDC$_s$ is defined as:

$$\text{FDC}_s = \frac{\sum_{i=1}^{n} (e_i - \overline{e})(d_i - \overline{d})}{\sqrt{\sum_{i=1}^{n} (e_i - \overline{e})^2} \sqrt{\sum_{i=1}^{n} (d_i - \overline{d})^2}}$$

where n is the size of a uniform sample of weight vectors, $\mathbf{W} = \{\mathbf{w}_1, ..., \mathbf{w}_n\}$, with associated error values $E = \{e_1, ..., e_n\}$; $\overline{e}$ is the mean of E, d_i is the Euclidean distance from $\mathbf{w}_i$ to the weight vector in the sample with the lowest error value, and $\overline{d}$ is the mean of all d_i.

The range of the FDC$_s$ measurement is $[-1, 1]$. For minimisation problems, a value close to 1 indicates a highly searchable landscape, a value close to 0 shows a lack of information in the landscape, and a negative value indicates a deceptive search landscape.

D. Information landscape negative searchability measure

Another measure of "problem hardness" was proposed by Borenstein and Poli [20]. In [20], an "information landscape" of a problem is generated by taking a random sample of the search space and performing pairwise fitness comparisons between the sampled points. To evaluate the amount and quality of information in the given landscape, the difference between the information landscape of the problem and the information landscape of an "optimal" landscape is calculated.

Malan [10] proposed an information landscape negative searchability (IL$_{ns}$) measure based on the Borenstein and Poli [20] approach. In [10], a random sample R is generated, and its information landscape is calculated. The spherical function is chosen as the "optimal" landscape, as it remains robustly searchable when scaled to higher dimensions. The spherical function is shifted such that its minimum coincides with the best solution in R. The difference between the information landscape of the problem on sample R and the information landscape of the spherical function on sample R is reported as the IL$_{ns}$ value.

IL$_{ns}$ essentially measures the distance of the given fitness landscape from the spherical function fitness landscape of the same dimensionality. IL$_{ns}$ is bounded to $[0, 1]$, where a value of 0 indicates maximum search information (no difference between the optimal landscape and the actual landscape), and a value of 1 indicates poor quality and quantity of the information.

The next section describes how FLA metrics apply in the NN context.

III. Error Landscapes of Neural Networks

In fully-connected feed-forward NN architectures, each neuron in a layer is connected to every neuron in the next layer, and each connection bears a weight. Given m weights, the solution space of all possible classifiers for a NN is a m-dimensional space of all possible weight vectors. Some of these weight vectors may yield a poor measure of error, while some may be good. The complete search space of all possible

NN weight vectors with associated error values is referred to as the "error landscape" of a NN in this study.

NN error landscapes have been investigated before from various perspectives. Gallagher [6] used techniques such as principal component analysis to simplify error landscape representation in order to visualise NN error landscapes. It was determined that error landscapes have many flat areas with sudden cliffs or ravines. It was also theoretically proved using random matrix theory that NN error landscapes exhibit more saddle points than local minima, and that the number of local minima diminishes exponentially as the dimensionality of the problem increases [4]. Choromanska *et al* [21] have drawn theoretical parallels between NN error landscapes and spin-glass models, once again concluding that saddle points are a more prominent feature of a NN error landscape than local optima. Malan and Engelbrecht's FLA metrics have been successfully applied to investigate the effect of multiple hidden layers on the resulting NN error landscapes [22]. However, Rakitianskaia *et al* [22] have only considered FLA measurements taken on the $[-1, 1]$ interval for every weight.

Error landscapes of NNs are in many ways similar to fitness landscapes of continuous optimisation problems, but there is an important difference: NN error landscapes are unbounded, whereas optimisation problems usually have bounded decision variables. NN weights do not have any meaning by themselves, and can be any numbers in $\mathbb{R}^m$, as opposed to decision variables in optimisation problems that relate to some limited resource in the real world.

The unbounded search spaces of NNs pose a problem to FLA, since the sampling algorithms used by FLA metrics, such as the progressive random walk [16] and the Manhattan random walk [14], require knowledge of the minimum and the maximum values per dimension. A range also needs to be specified to generate a random sample for the FDC$_s$ and the IL$_{ns}$ metrics.

No such range is defined for NNs. It is known, however, that NN weights are usually initialised in a small range around zero [12]. One reason behind choosing a small range is the avoidance of preliminary saturation. Saturation is the phenomenon when the hidden units of a NN predominantly output values close to the asymptotic ends of the activation function range. Indeed, if very large weights are used, the weighted sum of inputs is likely to have a large magnitude, causing the bounded activation functions to output near-asymptotic values. Saturated units make gradient descent learning slow and inefficient due to small derivative values near the asymptotes [23]. It was also shown that non-gradient descent learning can be hindered by NN saturation [24].

Therefore, it is not unreasonable to study the NN error landscapes on a small area around the origin, since that is exactly where the search for a solution begins. The search space, however, is unbounded, therefore the properties of the error landscape on a larger scale may provide insight into the dynamics of a training algorithm and the complexity of the problem. Performing FLA on large and small subsets of the search space would also demonstrate how FLA metrics scale, and whether the metrics converge to a similar value for different problems when the scale is increased. The purpose of this study was to perform FLA on a selection of NN problems, under various search space boundaries, and to determine the relationship between the FLA metrics and the search space boundaries imposed.

IV. Experimentation

This section details the experimental set-up used in this study. Section IV-A outlines the benchmark problems and the corresponding NN architectures used. Section IV-B describes the selection of search space boundaries tested.

A. Benchmark problems

Four classification benchmark problems outlined in Table I were used in this study. References next to the data set titles indicate the sources from which the data was borrowed. Each input and hidden layer had a bias unit, set to -1. All NNs employed the identity activation function in the input layer, and the sigmoid activation function in the hidden and output layers, defined as $f(x) = 1/(1 + e^{-x})$, where x is the net input signal. All input values were scaled to $[-1, 1]$ interval to lie within sigmoid's active domain, and the binary target values were scaled to $t_k \in \{0.1, 0.9\}$ to lie within sigmoid's output range, $(0, 1)$.

TABLE I
Benchmark Problems

Problem	In	Hidden	Out	**Dimensionality**
Iris [25]	4	2	3	**19**
Diabetes [26]	8	6	2	**68**
Glass [26]	9	9	6	**150**
Heart [26]	32	6	1	**205**

For the purpose of this study, the mean squared error (MSE) was used as the NN error measurement, given by:

$$E_{mse} = \frac{\sum_{p=1}^{P} \sum_{k=1}^{K} (t_{kp} - o_{kp})^2}{PK}$$

where K is the total number of outputs per pattern, and P is the total number of patterns. The MSE quantifies the magnitude of the distance between the generated outputs, o_{kp}, and target outputs, t_{kp}. The aim of training algorithms is to minimise MSE, i.e. minimise the distance between the outputs and the targets. The fitness landscape generated by MSE was analysed under a selection of boundaries discussed in the next section.

B. Search space boundaries

From the MSE perspective, the search space is infinite. However, from the perspective of a training algorithm, only a subspace of the infinite search space is ever traversed. Therefore, the part of the search space actually visited by a training algorithm has the most practical significance. The question is, what part of the search space does a training algorithm typically visit, and how much does the visited subspace vary per problem and per algorithm?

According to [12], the interval defined by $[-fanin^{-1/2}, fanin^{-1/2}]$, where $fanin$ is the number of connections leading into the node, is a good interval for weight initialisation, as it avoids saturation at early stages of training. For the architectures shown in Table I, $fanin^{-1/2}$ varies from 0.17 (heart) to 0.58 (iris). The larger the architecture, the smaller the weight initialisation range will be, and vice versa, but in all cases, $fanin^{-1/2} \leq 1$.

The weights are thus typically initialised within the $[-1, 1]$ interval. Do the training algorithms ever leave this interval? The easiest way to determine this is to train a NN and observe the resulting distribution of the weights. Figure 1 illustrates the weight distributions after 1000 iterations of 30 runs of the stochastic backpropagation algorithm with a learning rate of 0.1 and a momentum of 0.9 on the four problems listed in Table I. All weights were initialised in the corresponding $[-fanin^{-1/2}, fanin^{-1/2}]$ intervals, but, as Figure 1 shows, the final weights lay within the $[-15, 15]$ interval, a significantly wider interval than the initial interval. Thus, the initial $[-1, 1]$ interval is not representative enough for the purposes of FLA.

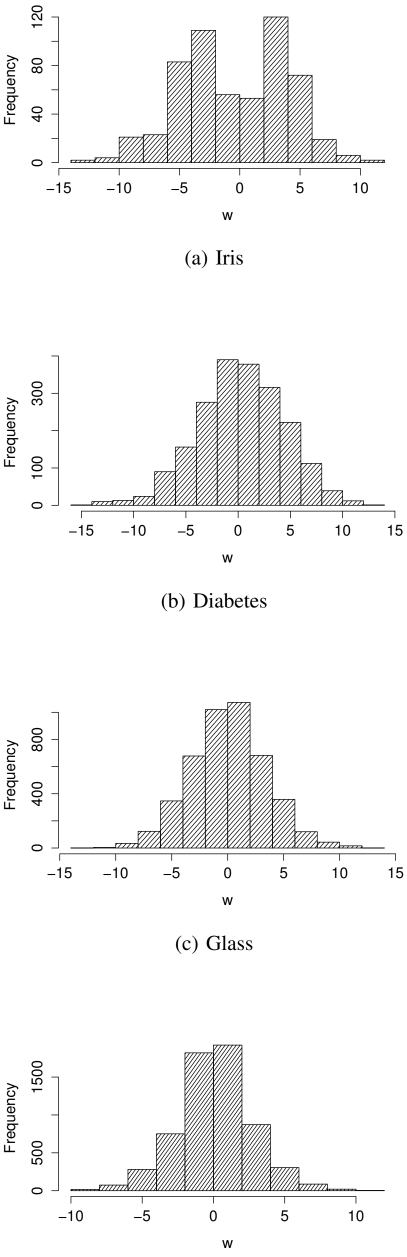
Backpropagation is indeed not the only training algorithm used in practice. In [27], a selection of particle swarm optimisation (PSO) algorithms was used to train NNs, and it was shown that a non-regularised PSO produces weights in the $[-200, 200]$ interval for the iris problem. It has also been shown in [28] that PSO tends to diverge on NN training problems, producing very large weights. Perhaps the shape of the error landscape is one of the reasons behind such divergent behaviour.

Another important property of NN error landscapes is their inherent symmetry [29]. Various permutations of hidden neurons in a layer yield identical NN models [29]. Flipping the signs of all the incoming and outgoing weights of a single neuron will also leave the neuron's output unchanged [29]. Reducing the search space to an asymmetric subspace can thus yield a less redundant search subspace, potentially easier to search.

Based on all the insights above, a selection of intervals for random sampling for the FLA metrics, both symmetric and asymmetric about the origin, was chosen for this study. The intervals used were $[-N, N], N \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$, and $[0, N], N \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$.

V. Experimental Results

Experimental results obtained under various search space boundaries for the four problems considered are presented in this section. The six FLA metrics mentioned in Section II were used to analyse the NN error landscapes with respect to MSE. Section V-A discusses the gradient measures. Section V-B discusses the ruggedness measures. Section V-C discusses the searchability measures. All reported results are averages over 30 independent runs.



(a) Iris

(b) Diabetes

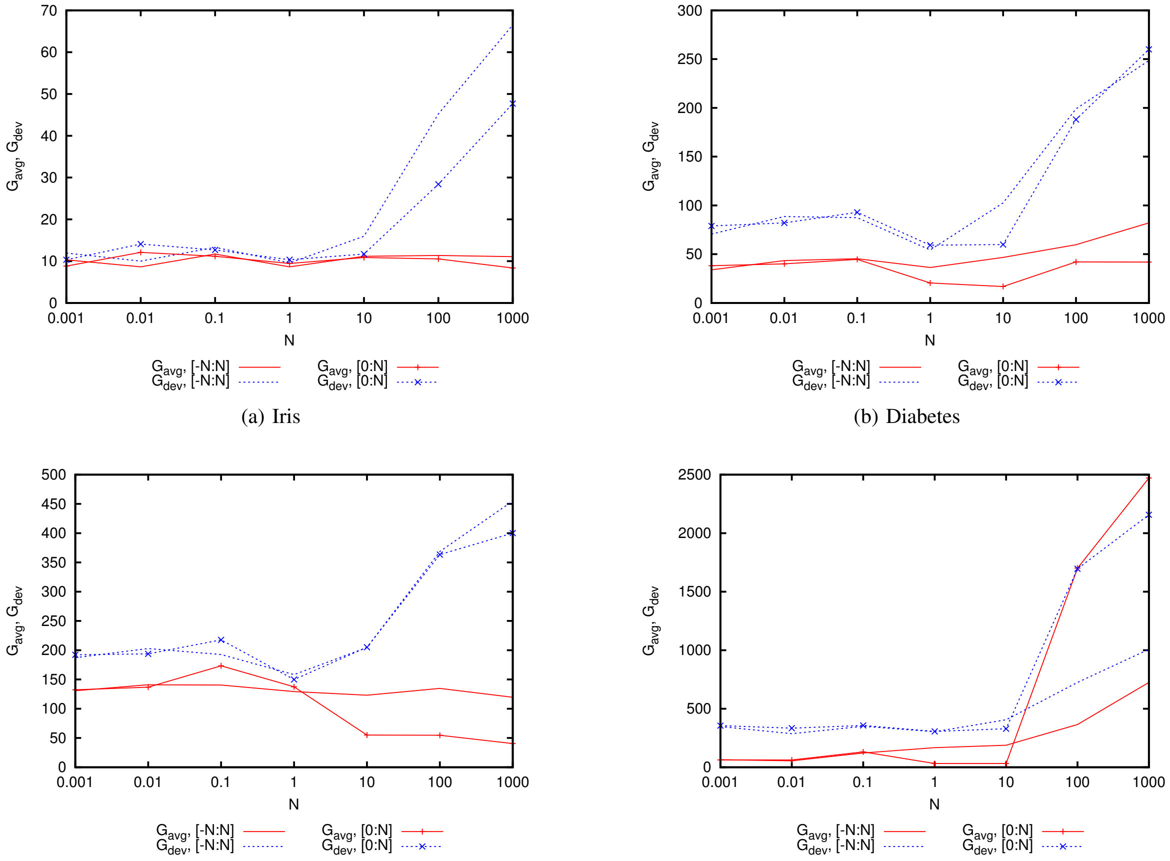
(c) Glass

(d) Heart

Fig. 1. Gradients

A. Gradients

Figure 2 shows the G_{avg} and G_{dev} measures obtained under different search space boundaries. The first notable feature of the gradient metrics is that even inside the smallest bounds ($[-0.001, 0.001]$, $[0, 0.001]$), reasonably large G_{avg} and G_{dev} were obtained for all problems. This result can be explained by the presence of a staircase-like, or "layered" structure of the error landscape, with sudden jumps from one layer to the next, as previously described in the literature [6], [7], [21]. The fact that very small intervals yield high gradients implies



(a) Iris

(b) Diabetes

(c) Glass

(d) Heart

Fig. 2. Gradients

that the "layered" structure is not a result of neuron saturation, since saturation does not occur for such small weights.

The magnitude of gradients increased with an increase in problem dimensionality. Thus, the jumps between layers become more and more drastic as the dimensionality grows. This corresponds to previously made observations that increasing dimensionality in NNs increases the number of saddle points surrounded by high error plateaus in the error landscapes [4]. This observation is also confirmed by the growing gap between G_{avg} and G_{dev}, where $G_{dev} > G_{avg}$, associated with the dimensionality increase, as shown in Figure 2. Malan [10] theorised that $G_{dev} >> G_{avg}$ is indicative of landscapes with step-like, sudden fitness changes.

The gradient measures remain mostly consistent for symmetric and asymmetric bounds alike for all $N \leq 1$. This is in fact the recommended weight initialisation range. The picture changes as the boundaries are widened: G_{dev} goes steadily upwards on all problems considered. The corresponding G_{avg} remains stable (iris, glass) or also increases (diabetes, heart). Wider boundaries used for the gradient metrics imply that the size of the step in the random walk becomes larger, too. For the estimation of gradients, Manhattan random walks are used: the maximum step size is fixed to 1% of the search space, but at every step, only one randomly chosen dimension is incremented or decremented by the given step. Thus, G_{avg} and G_{dev} also indicate how fast the gradients change in reaction to a change in one random dimension only. As the boundaries increase, G_{dev} grows faster than G_{avg}, indicating high variance in G_{avg}. Thus, bigger steps through the search space, even if made in one dimension (i.e. weight) only, are likely to change the fitness of the solution drastically. This effect will be further enhanced in higher-dimensional NNs.

Asymmetric bounds yielded similar or lower gradients than symmetric bounds on all problems for $N \leq 10$, which can be attributed to the fact that a smaller subspace of the search space was considered. However, the results were not consistent on problems of higher dimensionality. For the glass problem (150 weights), G_{avg} evidently decreased on large asymmetric regions. Perhaps asymmetric regions contained a higher degree of neutrality, or plateaus, due to saturation. Indeed, if all weights are positive and potentially large ($\forall w \in [0, 1000]$), the likelihood of a large net input signal is higher, resulting in higher degree of saturation.

For the heart problem (205 weights), both G_{avg} and G_{dev} yielded much higher values on large asymmetric regions than on the corresponding symmetric regions. Even though the heart problem had more weights in total, it used only one neuron on the output layer. The glass problem, on the other hand, used 6 output neurons. Thus, the final error was calculated over six outputs for the glass problem, and over one output for the heart problem. It can be argued that with six outputs, the chances of output unit saturation are multiplied by six, yielding more plateaus.

B. Ruggedness

Figure 3 shows the FEM values obtained under different search space boundaries. For all problems considered, $\text{FEM}_{0.01}$ and $\text{FEM}_{0.1}$ were within the range $[0.2, 0.3]$ for all $N \leq 0.1$, indicating mostly non-rugged, consistent landscapes. The picture changed drastically as N increased from 0.1 to 1. For all problems considered, macro-ruggedness on the symmetric $[-1, 1]$ region exceeded 0.5, indicating a change from mostly uniform to mostly rugged. Corresponding asymmetric regions did not exhibit an increase in $\text{FEM}_{0.1}$, or exhibited a less drastic increase. Thus, weights with absolute values between $[0.1, 1]$ constitute a rich search landscape with high variability of fitness values. However, if the search space consists of positive weights only, the variability is greatly decreased. Therefore, asymmetric regions may have less information for NN training.

For all problems considered, both $\text{FEM}_{0.01}$ and $\text{FEM}_{0.1}$ increased as the symmetric search space widened, and $\text{FEM}_{0.1}$ produced larger values than $\text{FEM}_{0.01}$ at all times. Thus, the error landscapes were relatively smooth and consistent on the micro scale ($\text{FEM}_{0.01}$), but rather rugged on the macro scale ($\text{FEM}_{0.1}$). This can be attribted to the aforementioned layered structure of the NN error landscapes: little change is observed on a given "level", but a transition from one level to the next represents a significant change in fitness.

FEM characteristics of the symmetric regions were more consistent than that of the asymmetric regions. The instability of FEM observed on the asymmetric regions indicates that the asymmetric regions chosen did not provide a good representation of the error landscapes. Sudden dips in FEM may correspond to the neutralities dominating the saturated part of the search space. Shifting the search space region to the positive weights only also implied that some of the random walks would start from the origin, while symmetric regions guaranteed that the random walks would start on the outer boundaries of the selected regions. Inconsistencies in the asymmetric FEM values indicate that the view of the search space as seen from the origin is quite different from the view as seen from the boundaries of the search space. It would be interesting to design a training algorithm that starts the search on the boundaries, and gravitates towards the origin, somewhat similar to NN weight regularisation. Implementation of such learning strategy is left for future research.

C. Searchability

Figure 4 shows the FDC_s and IL_{ns} measures obtained under different search space boundaries. IL_{ns} consistently increased as the boundaries increased, indicating that wider search spaces contained less and/or poorer information to guide the search. This applied to both symmetric and asymmetric bounds. Indeed, a wider search space implies larger weights, and larger weights imply a higher degree of saturation, while saturated regions of the search space are known to be hard to search. The amount of information quantified by IL_{ns} also decreased as the dimensionality of the problem increased, rightfully indicating that higher-dimensional problems are harder to search.

FDC_s, similar to IL_{ns}, identified higher-dimensional problems as less searchable, which is to be expected. FDC_s decreased as the boundaries increased, and on most problems the transition from $N = 0.1$ to $N = 1$ yielded a drastic drop in searchability. It was previously observed that the same transition yielded a drastic increase in $\text{FEM}_{0.1}$ values. Thus, higher fitness variability corresponded to poorer searchability, and the regions identified as "more searchable" by the FDC_s may simply be more flat.

Symmetric regions were quantified as less searchable than the asymmetric regions by FDC_s. Indeed, the asymmetric regions were less redundant, and thus contained less variability. For higher-dimensional problems (glass, diabetes), the FDC_s measurements of the asymmetric regions strongly correlated with the corresponding gradient measures shown in Figure 2. Lower gradients resulted in lower searchability, while steep gradients were associated with high searchability. Indeed, an absence of gradients would leave a training algorithm with nothing to go on.

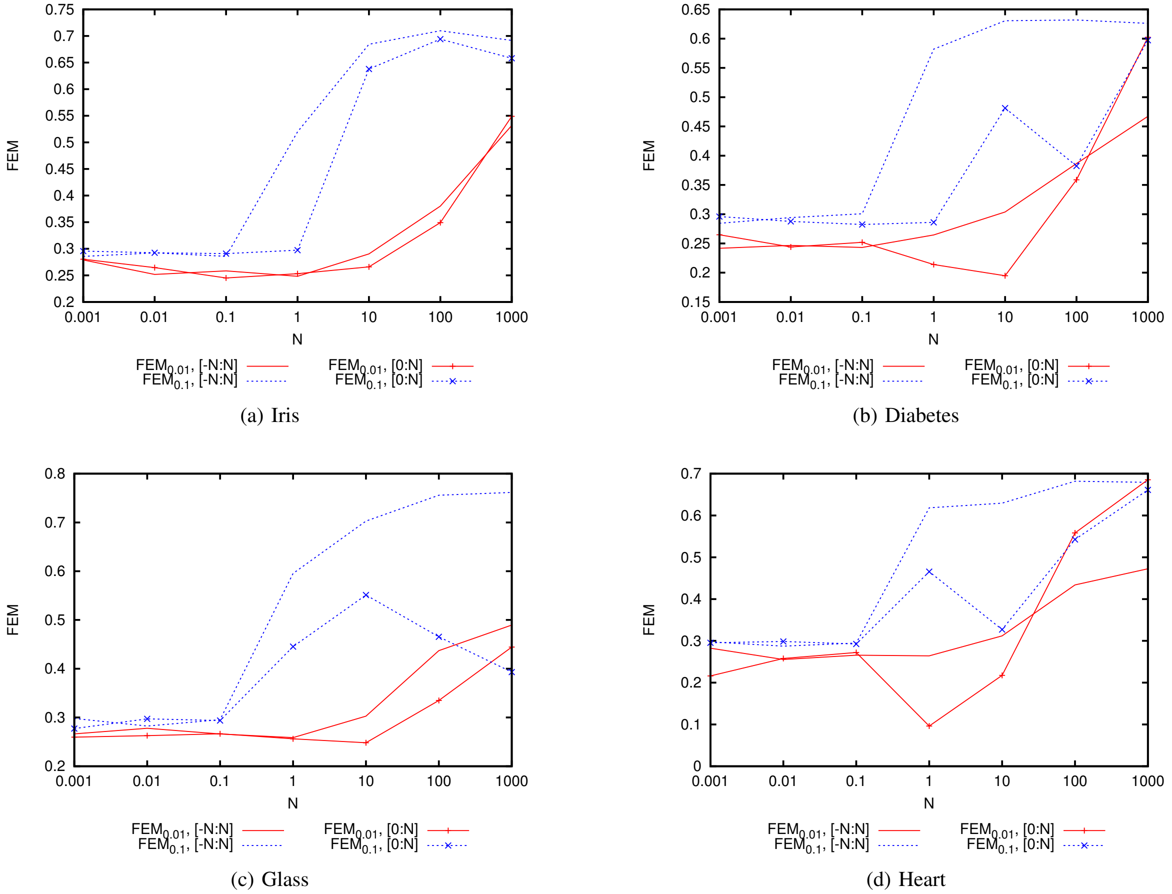
VI. Conclusion

This study investigated the behaviour of various FLA metrics on NN search spaces under different bounds. All FLA metrics used in this study exhibited a sensitivity to the bounds chosen. NNs generate complex error landscapes indeed, and the properties of landscapes observed on a small area around the origin do not apply to the entire unbounded search space.

High gradient values were obtained on both small and large search subspaces, indicating that steep gradients constitute an inherent NN error landscape property. Gradient magnitudes increased with increase in problem dimensionality. Increasing search space boundaries increased the variance of gradients, indicating that away from the origin, the step-like jumps between plateaus become more and more drastic.

According to the ruggedness metric, the NN error landscapes exhibited very little variation in $[-0.1, 0.1]$ region, but the entropy increased drastically for weights with absolute values within $[0.1, 1]$. It is worth exploring the potential of this interval for weight initialisation. For increased search space boundaries, the micro-ruggedness increased slower than the macro-ruggedness, indicating relevant consistency for small steps through the search space, and more drastic changes for larger steps.

Searchability metrics indicated a decrease in searchability associated with the increase of search space boundaries and dimensionality. Asymmetric regions appeared less steep, less rugged, and more searchable than the symmetric regions. This



(a) Iris

(b) Diabetes

(c) Glass

(d) Heart

Fig. 3. FEM

behaviour is attributed to a higher saturation degree exhibited by all-positive weights, as well as a lower level of optima redundancy. Evaluating the maximum fitness of the weight vectors in asymmetric subspaces is left for future research.

The FLA measures for NN error landscapes clearly depend on the search space boundaries chosen. Based on the observations made in this study, a range of regions rather than a single region should be used if FLA properties of a NN are to be studied. The two suggested regions are the region in which weights are initialised, as well as the region explored by the training algorithm of choice. More regions can be added to gain more insight into the problem.

Larger regions of the search space were classified as highly rugged, with extremely steep gradients and little information to guide a training algorithm. This explains why weight-dampening techniques such as regularisation are so effective. In general, it is desirable to make training algorithms gravitate towards the origin while allowing exploration. Development of such gravitational approach in the particle swarm optimisation context is a topic of future research.

References

[1] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, UK: Oxford University Press, 1995.

[2] G. Dreyfus, *Neural networks: methodology and applications*. Berlin, Germany: Springer, 2005.

[3] A. Choromanska, Y. LeCun, and G. B. Arous, “Open problem: The landscape of the loss surfaces of multilayer networks,” in *Proceedings of The 28th Conference on Learning Theory*, 2015, pp. 1756–1760.

[4] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization,” in *Advances in Neural Information Processing Systems*, 2014, pp. 2933–2941.

[5] L. G. Hamey, “XOR has no local minima: A case study in neural network error surface analysis,” *Neural Networks*, vol. 11, no. 4, pp. 669–681, 1998.

[6] M. R. Gallagher, “Multi-layer perceptron error surfaces: Visualization, structure and modelling,” Ph.D. dissertation, University of Queensland, St Lucia 4072, Australia, 2000.

[7] D. R. Hush, B. Horne, and J. M. Salas, “Error surfaces for multilayer perceptrons,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 5, pp. 1152–1161, Oct. 1992.

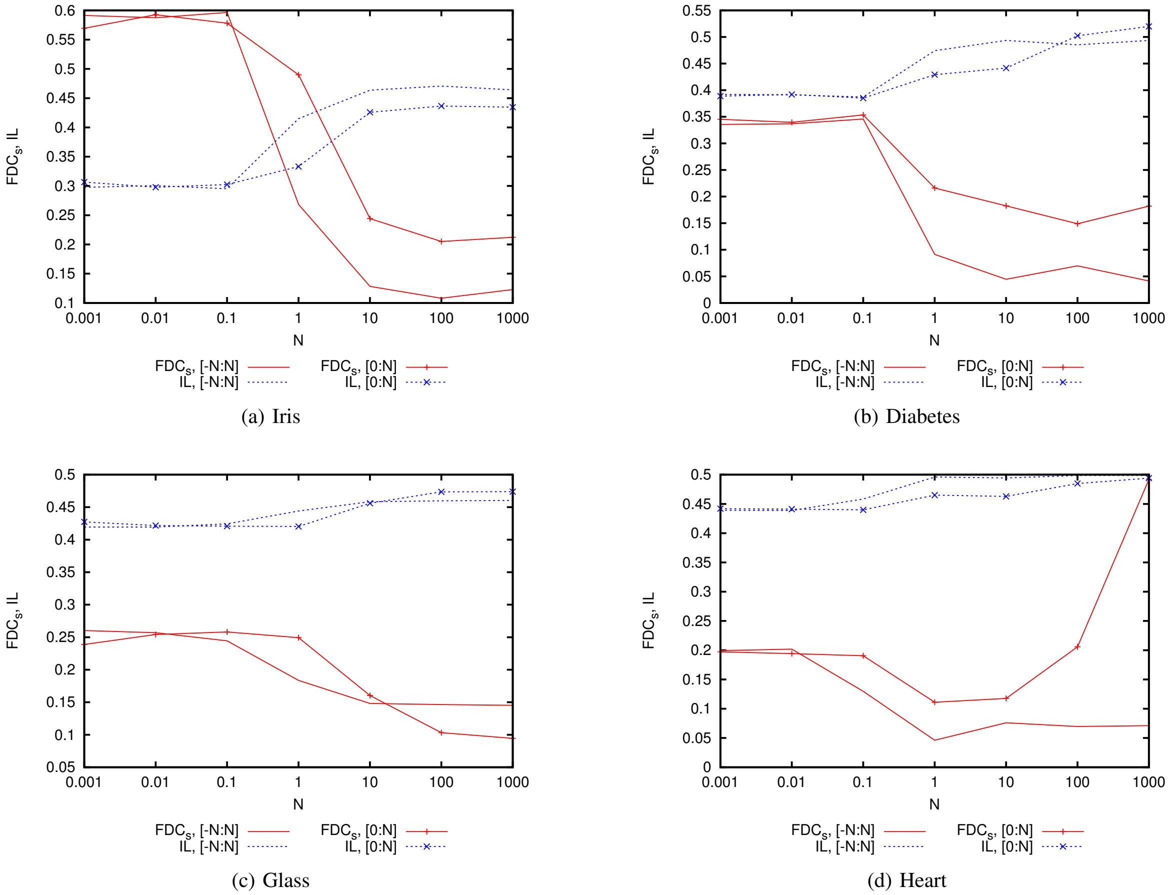
[8] T. Jones, “Evolutionary algorithms, fitness landscapes and search,” Ph.D. dissertation, The University of New Mexico, 1995.

[9] P. Merz and B. Freisleben, “Fitness landscape analysis and memetic algorithms for the quadratic assignment problem,” *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 4, pp. 337–352, 2000.

[10] K. M. Malan, “Characterising continuous optimisation problems for particle swarm optimisation performance prediction,” Ph.D. dissertation, University of Pretoria, 2014.

[11] E. Pitzer and M. Affenzeller, “A comprehensive survey on fitness landscape analysis,” in *Recent Advances in Intelligent Engineering Systems*. Springer, 2012, pp. 161–191.

[12] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, “Efficient



(a) Iris

(b) Diabetes

(c) Glass

(d) Heart

Fig. 4. FDC and IL

backprop," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 9–48.

[13] K. M. Malan and A. P. Engelbrecht, "A survey of techniques for characterising fitness landscapes and some possible ways forward," *Information Sciences*, vol. 241, pp. 148–163, 2013.

[14] ——, "Ruggedness, funnels and gradients in fitness landscapes and the effect on PSO performance," in *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 963–970.

[15] V. K. Vassilev, T. C. Fogarty, and J. F. Miller, "Smoothness, ruggedness and neutrality of fitness landscapes: from theory to application," in *Advances in evolutionary computing*. Springer, 2003, pp. 3–44.

[16] K. M. Malan and A. P. Engelbrecht, "A progressive random walk algorithm for sampling continuous fitness landscapes," in *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2014, pp. 2507–2514.

[17] ——, "Quantifying ruggedness of continuous landscapes using entropy," in *IEEE Congress on Evolutionary Computation*. IEEE, 2009, pp. 1440–1447.

[18] T. Jones and S. Forrest, "Fitness distance correlation as a measure of problem difficulty for genetic algorithms," in *Proceedings of the 6th International Conference on Genetic Algorithms*. Morgan Kaufmann Publishers Inc., 1995, pp. 184–192.

[19] K. M. Malan and A. P. Engelbrecht, "Characterising the searchability of continuous optimisation problems for PSO," *Swarm Intelligence*, vol. 8, no. 4, pp. 275–302, 2014.

[20] Y. Borenstein and R. Poli, "Information landscapes," in *Proceedings of the 7th annual conference on genetic and evolutionary computation*. ACM, 2005, pp. 1515–1522.

[21] A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, "The loss surfaces of multilayer networks," in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, 2015, pp. 192–204.

[22] A. Rakitianskaia, E. Bekker, K. Malan, and A. Engelbrecht, "Analysis of error landscapes in multi-layered neural networks for classification," in *Proceedings of the IEEE Congress on Evolutionary Computation*. Vancouver, Canada: IEEE, 2016, in press.

[23] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[24] A. Rakitianskaia and A. Engelbrecht, "Saturation in PSO neural network training: Good or evil?" in *Proceedings of the IEEE Congress on Evolutionary Computation*. Sendai, Japan: IEEE, 2015, pp. 125–132.

[25] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936. [Online]. Available: http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x

[26] L. Prechelt, "Proben1: A set of neural network benchmark problems and benchmarking rules," Fakultät für Informatik, Universität Karlsruhe, Karlsruhe, Germany, Tech. Rep. 21/94, September 1994.

[27] A. Rakitianskaia and A. Engelbrecht, "Weight regularisation in particle swarm optimisation neural network training," in *Proceedings of the IEEE Symposium on Swarm Intelligence*. Florida, USA: IEEE, 2014, pp. 1–8.

[28] A. B. Van Wyk and A. P. Engelbrecht, "Overfitting by PSO trained feedforward neural networks," in *Proceedings of the IEEE Congress on Evolutionary Computation*, 2010, pp. 1–8.

[29] A. M. Chen, H.-m. Lu, and R. Hecht-Nielsen, "On the geometry of feedforward neural network error surfaces," *Neural computation*, vol. 5, no. 6, pp. 910–927, 1993.