Additional material

Anonymous Author(s)
Affiliation
Address
email

Abstract

The purpose of this document is to share extra material with the interested reviewers. Firstly, the progressive gradient walk is discussed in more detail. The authors deemed necessary to include the algorithm description for the sake of highlighting the difference between the sampling used in the experiments and the standard stochastic gradient descent. Experiments were run on 7 different classification problems, but only two problems are reported in the final paper for the sake of brevity. Loss-gradient clouds for the remaining 5 problems are provided here for your interest. The same general trends can be observed across all problems.

1 Progressive gradient walk

A progressive gradient walk uses the numeric gradient of the loss function to determine the direction of each step. The size of the step is randomised per dimension within predefined bounds. The progressive gradient walk algorithm is summarised as follows:

1. Gradient vector $\vec{g}_l$ is calculated for a point $\vec{x}_l$.
2. A binary direction mask $\vec{b}_l$ is extracted from $\vec{g}_l$ as follows:

$$b_{lj} = \begin{cases} 0, & \text{if } g_{lj} < 0 \\ 1, & \text{otherwise} \end{cases}$$

   where $j \in \{1, \ldots, m\}$ for the m-dimensional vector $\vec{g}_l$.
3. The progressive random walk algorithm, proposed in [2], is used to generate the next step $\vec{x}_{l+1}$. A single step of a progressive random walk can be defined as randomly generating an m-dimensional step vector $\Delta\vec{x}_l$, such that $\Delta x_{lj} \in [0, \varepsilon] \; \forall j \in \{1, \ldots, m\}$, and setting the sign of each Δx_{lj} according to the corresponding b_{lj}:

$$\Delta x_{lj} = \begin{cases} -\Delta x_{lj}, & \text{if } b_{lj} = 0. \\ \Delta x_{lj}, & \text{otherwise.} \end{cases}$$

   To generate the next step, $\vec{x}_{l+1}$, the current step $\vec{x}_l$ is modified by adding $\Delta\vec{x}_l$:

$$\vec{x}_{l+1} = \vec{x}_l + \Delta\vec{x}_l$$

The progressive gradient walk algorithm requires one parameter to be set: the maximum dimension-wise step size, ε.

2 Additional problems

Table 1 summarises the NN architecture parameters used for the datasets not reported in the main paper, as well as the total dimensionality of the corresponding weight space. The specified sources point to publications from which each dataset and/or NN architectures were adopted.

Submitted to 33rd Conference on Neural Information Processing Systems (NeurIPS 2019). Do not distribute.

Table 1: Benchmark Problems

Problem	Input	h	Output	Dimension	Source
Iris	4	4	3	35	[1]
Diabetes	8	8	1	81	[3]
Glass	9	9	6	150	[3]
Cancer	30	10	1	321	[3]
Heart	32	10	1	341	[3]

The properties of each dataset are briefly discussed below:

1. **Iris:** The famous Iris flower data set [1] contains 50 specimens from each of the three species of iris flowers, i.e. *Iris Setosa*, *Iris Versicolor*, and *Iris Virginica*. There are 150 patterns in the dataset.
2. **Diabetes:** The diabetes dataset [3] captures personal data of 768 Pima Indian patients, classified as diabetes positive or diabetes negative.
3. **Glass:** The glass dataset [3] captures chemical components of glass shards. Each glass shard belongs to one of six classes: float processed or non-float processed building windows, vehicle windows, containers, tableware, or head lamps. There are 214 patterns in the dataset.
4. **Cancer:** The breast cancer Wisconsin (diagnostic) dataset [3] consists of 699 patterns, each containing tumor descriptors, and a binary classification into benign or malignant.
5. **Heart:** The heart disease prediction dataset [3] contains 920 patterns, each describing various patient descriptors.

Various visualisation obtained for the problems listed in Table 1 are presented below.

3 Iris

The following figures are provided for Iris:

1. Figure 1: sampled curvature summary for the various architectures.
2. Figure 2: l-g clouds for different hidden layer widths.
3. Figure 3: l-g clouds for different hidden layer widths, colourised according to E_g.
4. Figure 4: l-g clouds for different architecture depths.
5. Figure 5: l-g clouds for different architecture depths, colourised according to E_g.

4 Diabetes

The following figures are provided for Diabetes:

1. Figure 6: sampled curvature summary for the various architectures.
2. Figures 7, 8, and 9: l-g clouds for different hidden layer widths.
3. Figure 10: l-g clouds for different architecture depths.

5 Glass

The following figures are provided for Glass:

1. Figure 11: sampled curvature summary for the various architectures.
2. Figure 12: l-g clouds for the various architectures.

6 Cancer

The following figures are provided for Glass:

1. Figure 13: sampled curvature summary for the various architectures.
2. Figures 14 and 15: l-g clouds for the various architectures.

7 Heart

The following figures are provided for Glass:

1. Figure 16: sampled curvature summary for the various architectures.
2. Figures 17 and 18: l-g clouds for the various architectures.

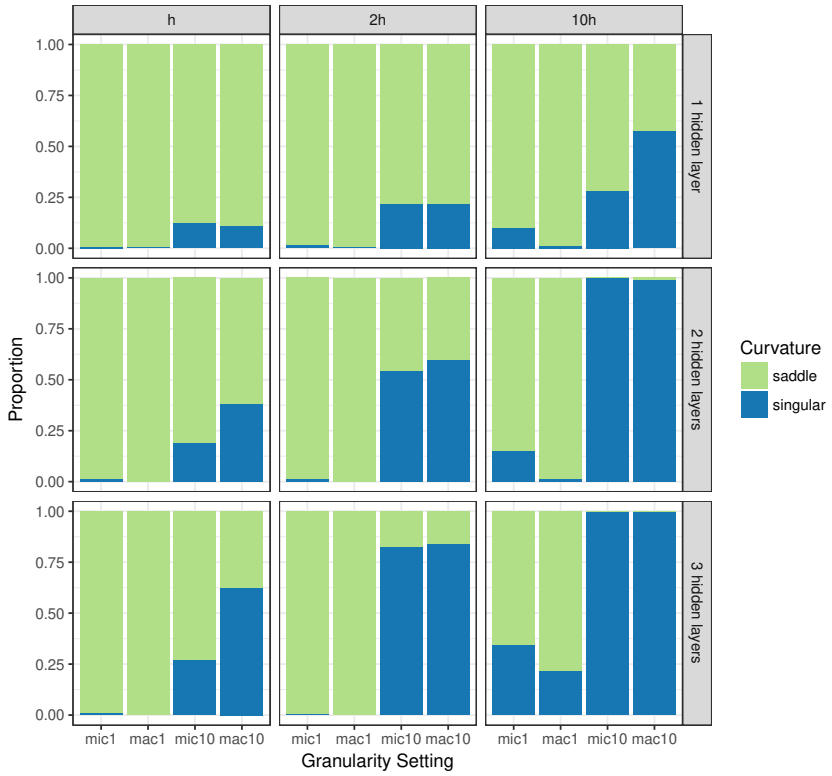


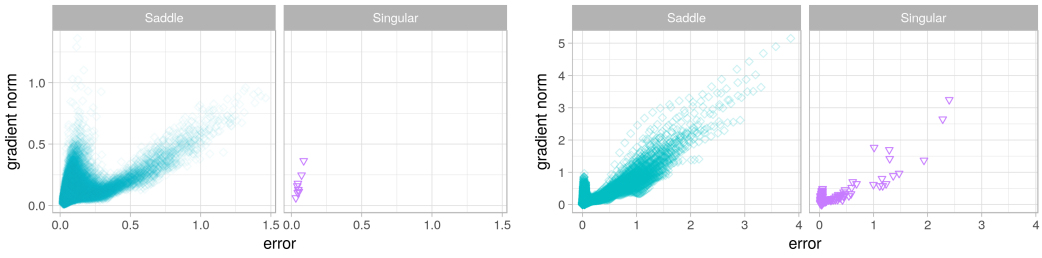
Figure 1: Histogram representation of the curvature information sampled by the gradient walks for the Iris problem for various NN architecture settings. An increase in either the hidden layer width, or the number of hidden layers, yielded increased landscape flatness.

References

[1] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

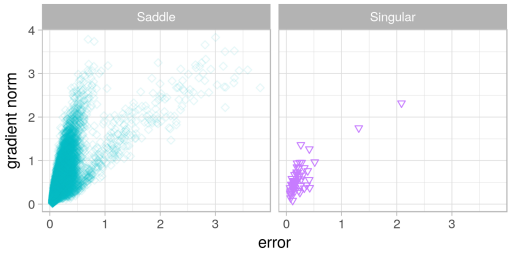
[2] Katherine M Malan and Andries P Engelbrecht. A progressive random walk algorithm for sampling continuous fitness landscapes. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 2507–2514. IEEE, 2014.

[3] Lutz Prechelt. Proben1 – a set of neural network benchmark problems and benchmarking rules. Technical report, Universität Karlsruhe, Karlsruhe, Germany, September 1994.

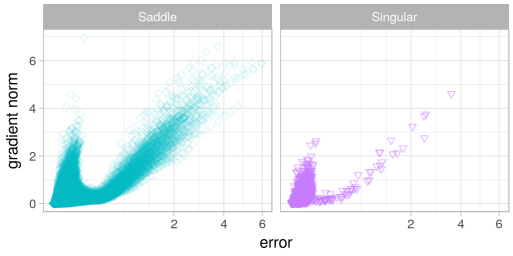


(a) $h = 4$, macro, $[-1, 1]$, 1 layer

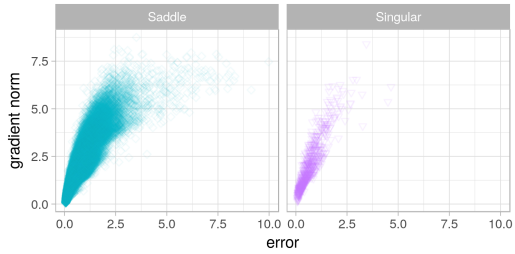
(b) $h = 4$, micro, $[-1, 1]$, 2 layers



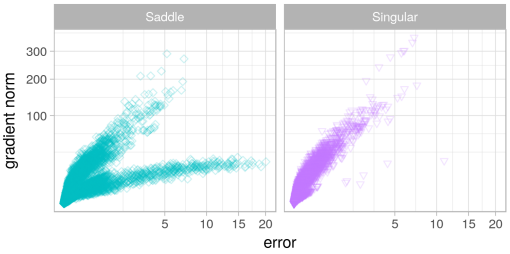
(c) $2 \times h = 8$, macro, $[-1, 1]$, 1 layer



(d) $2 \times h = 8$, micro, $[-1, 1]$, 2 layers

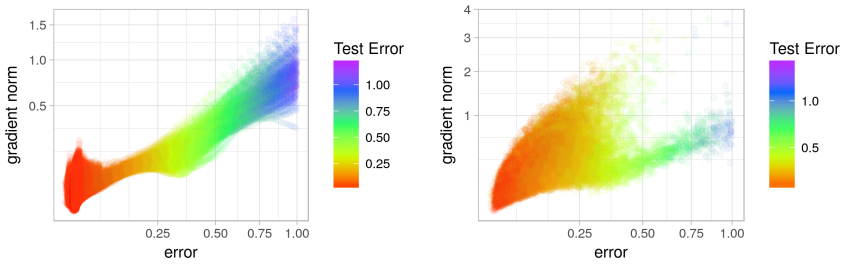


(e) $10 \times h = 40$, macro, $[-1, 1]$, 1 layer



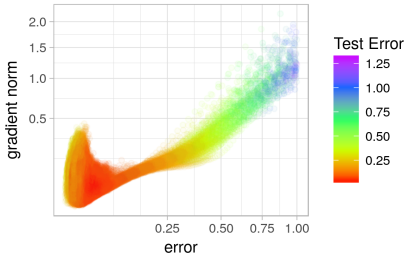
(f) $10 \times h = 40$, micro, $[-1, 1]$, 2 layers

Figure 2: L-g clouds for the micro and macro gradient walks initialised in the $[-1, 1]$ range for the Iris problem for the various number of hidden neurons in 1- and 2-hidden layer NNs. The total number of differently shaped attractors reduced with an increase in the hidden layer width.

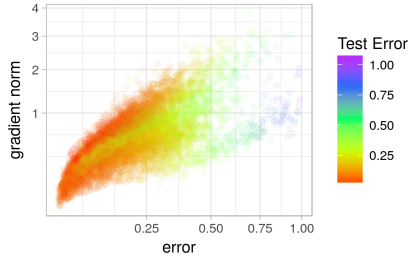


(a) $h = 4$, micro, $[-1, 1]$, $E_t < 1$

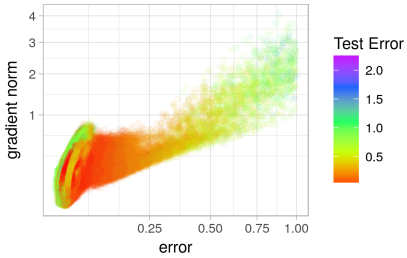
(b) $h = 4$, macro, $[-1, 1]$, $E_t < 1$



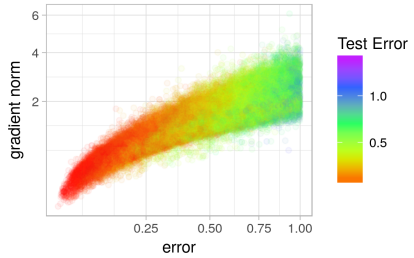
(c) $2 \times h = 8$, micro, $[-1, 1]$, $E_t < 1$



(d) $2 \times h = 8$, macro, $[-1, 1]$, $E_t < 1$

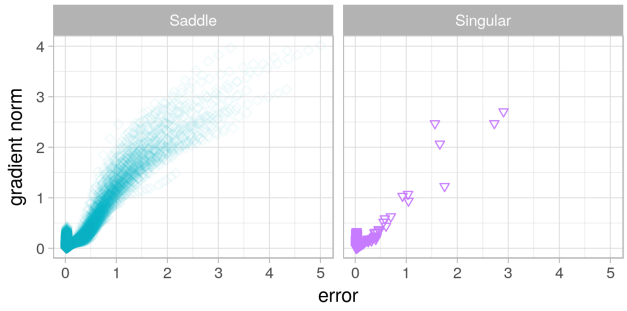


(e) $10 \times h = 40$, micro, $[-1, 1]$, $E_t < 1$

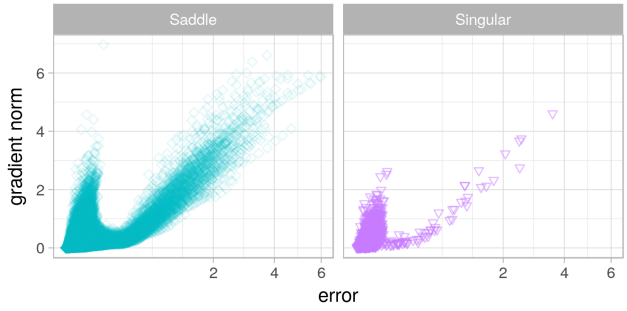


(f) $10 \times h = 40$, macro, $[-1, 1]$, $E_t < 1$

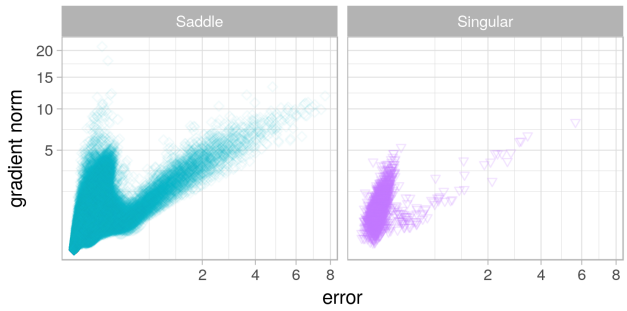
Figure 3: L-g clouds colourised according to the corresponding E_g values, obtained for single hidden layer NN architectures of varied hidden layer size for the Iris problem. Deterioration of the generalisation quality around the global minimum attractor associated with the increased hidden layer size is evident for the micro walks.



(a) 1 hidden layer, micro, $[-1, 1]$

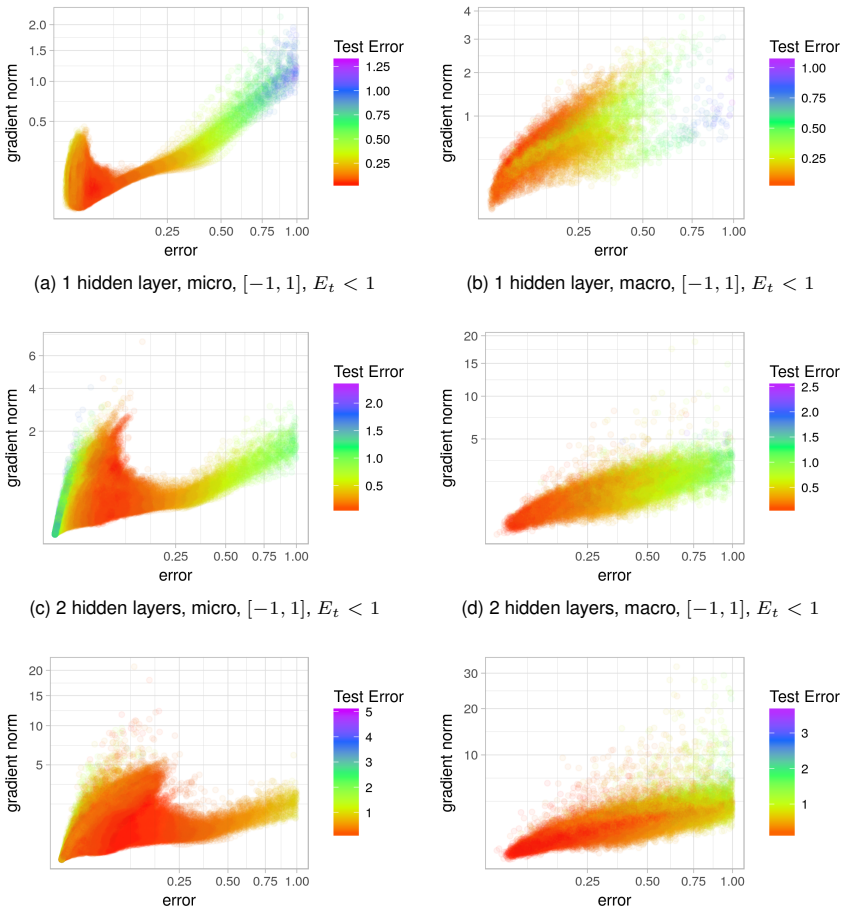


(b) 2 hidden layers, micro, $[-1, 1]$



(c) 3 hidden layers, micro, $[-1, 1]$

Figure 4: L-g clouds for the micro gradient walks initialised in the $[-1, 1]$ range for the Iris problem for the various number of hidden layers, with $h = 8$ for each layer. An increase in the number of hidden layers yielded a heavier steep gradient cluster. The steep gradient cluster overlapped with indefinite, i.e. flat points, attributed to the embedded minima.



(a) 1 hidden layer, micro, $[-1, 1]$, $E_t < 1$

(b) 1 hidden layer, macro, $[-1, 1]$, $E_t < 1$

(c) 2 hidden layers, micro, $[-1, 1]$, $E_t < 1$

(d) 2 hidden layers, macro, $[-1, 1]$, $E_t < 1$

(e) 3 hidden layers, micro, $[-1, 1]$, $E_t < 1$

(f) 3 hidden layers, macro, $[-1, 1]$, $E_t < 1$

Figure 5: L-g clouds colourised according to the corresponding E_g values, obtained for the various number of hidden layers, with $h = 8$ for each layer on the Iris problem. The range of E_g values corresponding to the points with $E_t \in [0, 1]$ increased as more hidden layers were added.

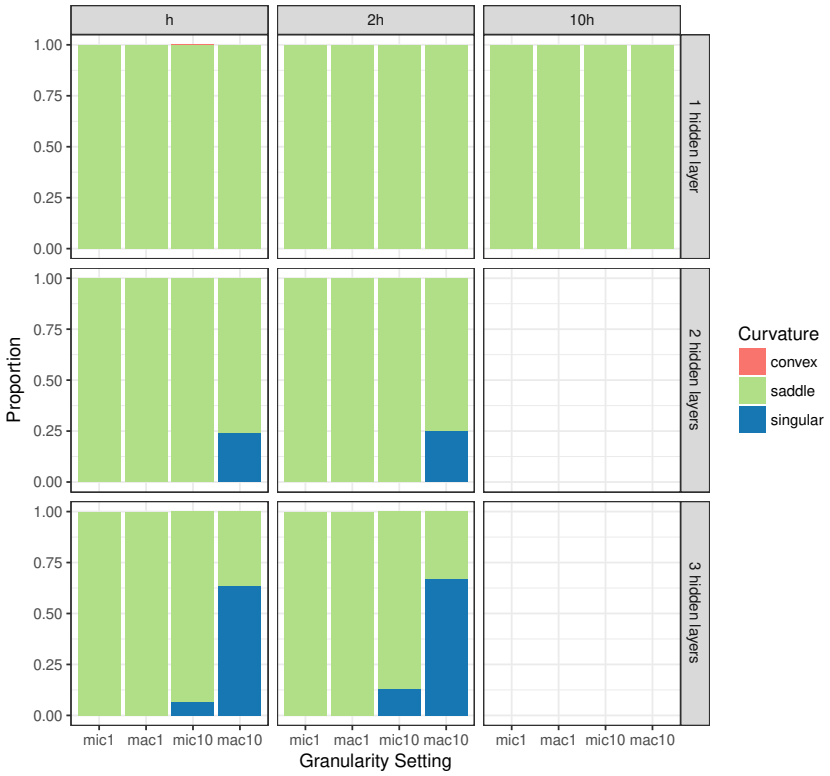
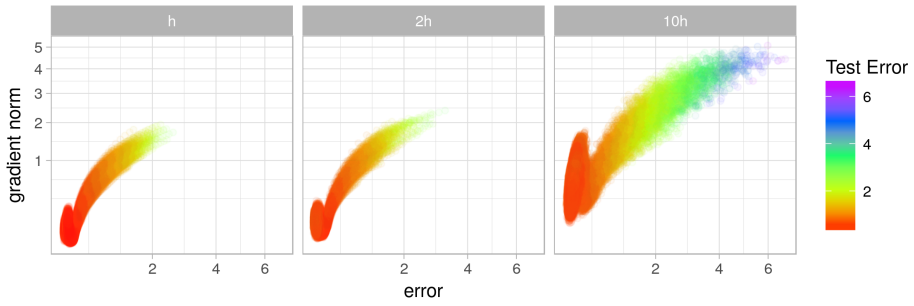
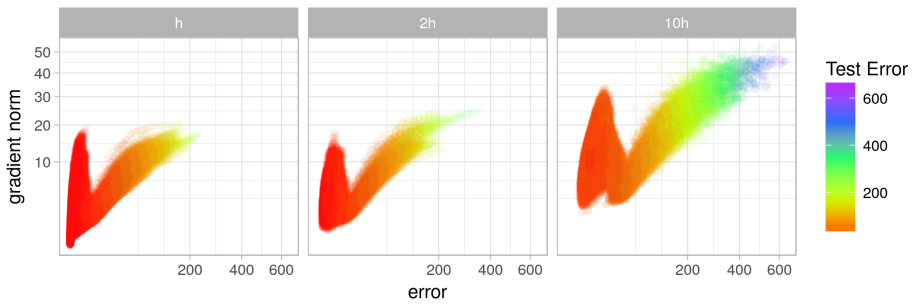


Figure 6: Histogram representation of the curvature information sampled by the gradient walks for the Diabetes problem for various NN architectures.

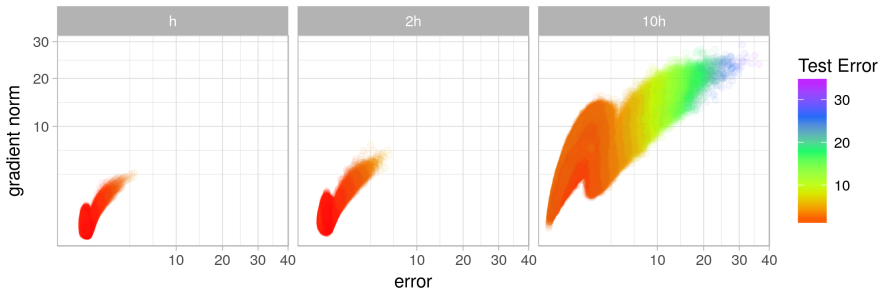


(a) $[-1, 1]$, micro

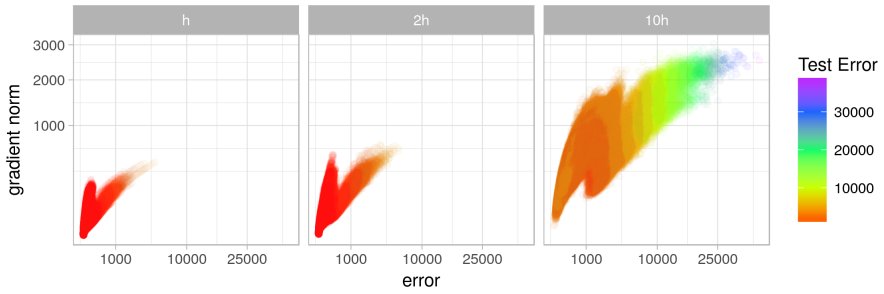


(b) $[-10, 10]$, micro

Figure 7: L-g clouds for the micro gradient walks initialised in the $[-1, 1]$ and $[-10, 10]$ ranges for the Diabetes problem for the various number of hidden neurons in a single hidden layer. The attractor widened as the number of hidden neurons increased.

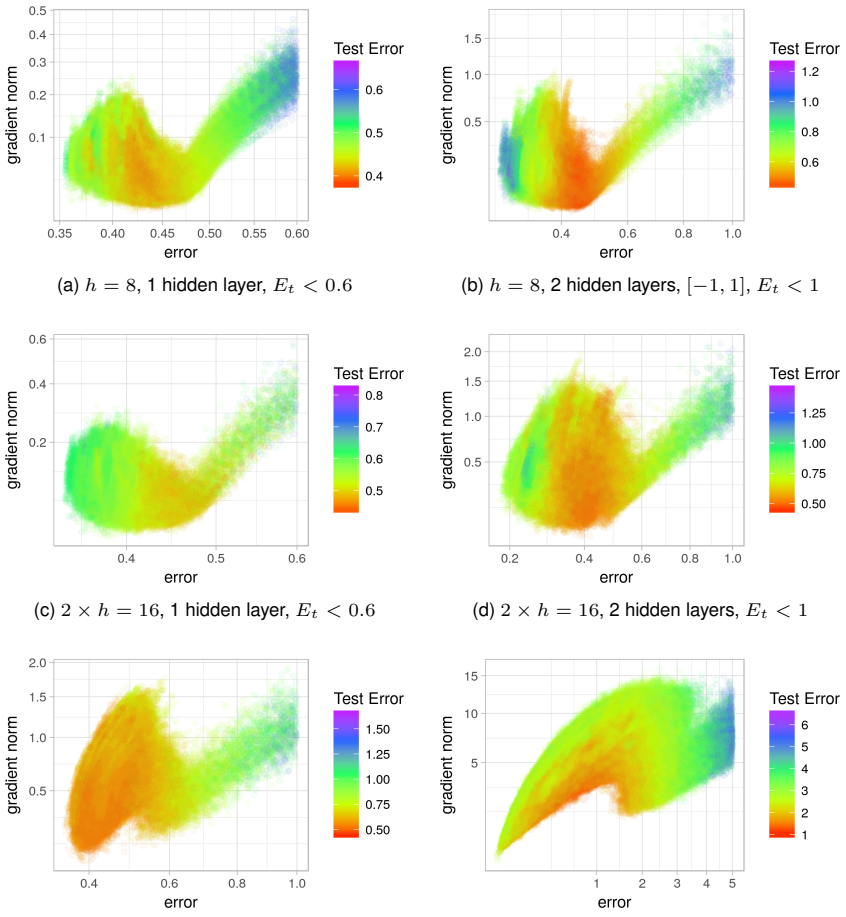


(a) $[-1, 1]$, micro



(b) $[-10, 10]$, micro

Figure 8: L-g clouds for the micro gradient walks initialised in the $[-1, 1]$ and $[-10, 10]$ ranges for the Diabetes problem for the various number of hidden neurons in two consecutive hidden layers. An increase in the hidden layer size exaggerated the split into two attractors for the 2-layer architectures.



(a) $h = 8$, 1 hidden layer, $E_t < 0.6$

(b) $h = 8$, 2 hidden layers, $[-1, 1]$, $E_t < 1$

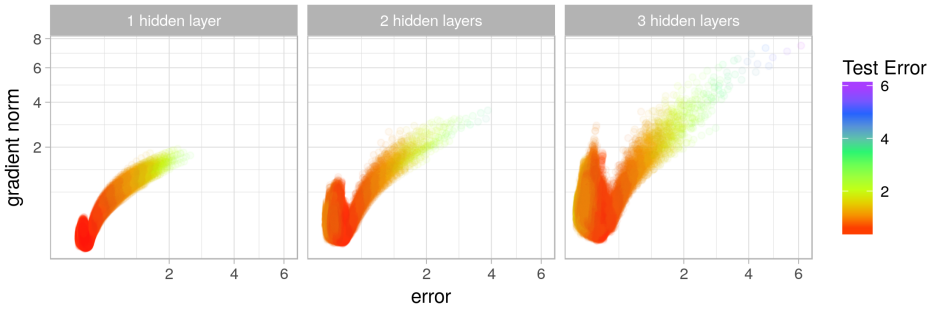
(c) $2 \times h = 16$, 1 hidden layer, $E_t < 0.6$

(d) $2 \times h = 16$, 2 hidden layers, $E_t < 1$

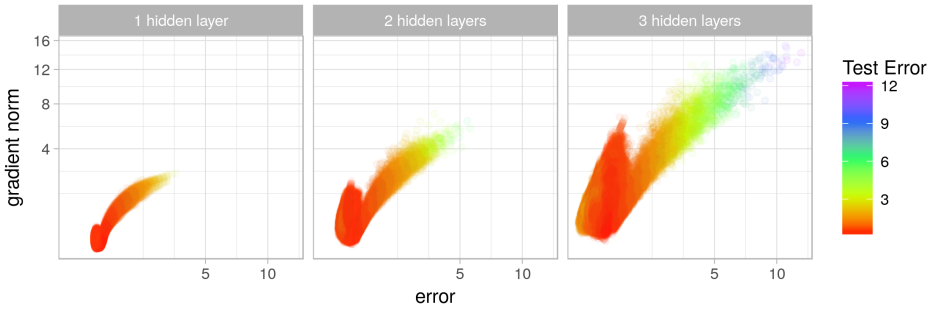
(e) $10 \times h = 80$, 1 hidden layer, $E_t < 1$

(f) $10 \times h = 80$, 2 hidden layers, $E_t < 5$

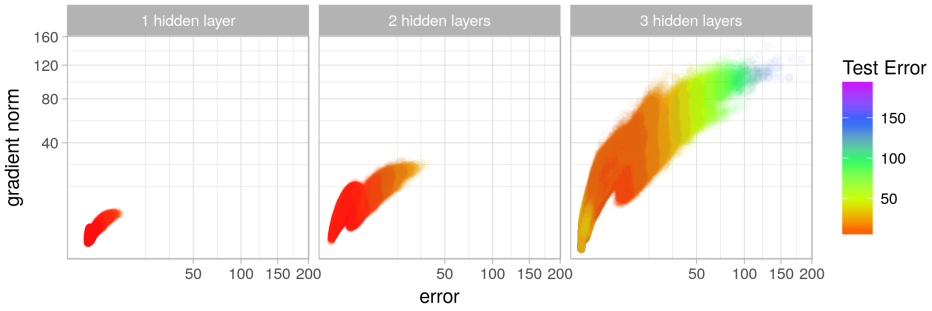
Figure 9: L-g clouds colourised according to the corresponding E_g values, obtained by the $[-1, 1]$ micro walks for the Diabetes problem. Exploitation of the global minimum attractor had a detrimental effect on generalisation, but the detrimental effect weakened as more hidden neurons were added.



(a) $h = 8$



(b) $2 \times h = 16$



(c) $10 \times h = 80$

Figure 10: L-g clouds for the micro gradient walks initialised in the $[-1, 1]$ range for the Diabetes problem for the various number of hidden layers. The shape of the attractors did not change with the addition of more layers, but rather widened in terms of the error and gradient magnitude ranges.

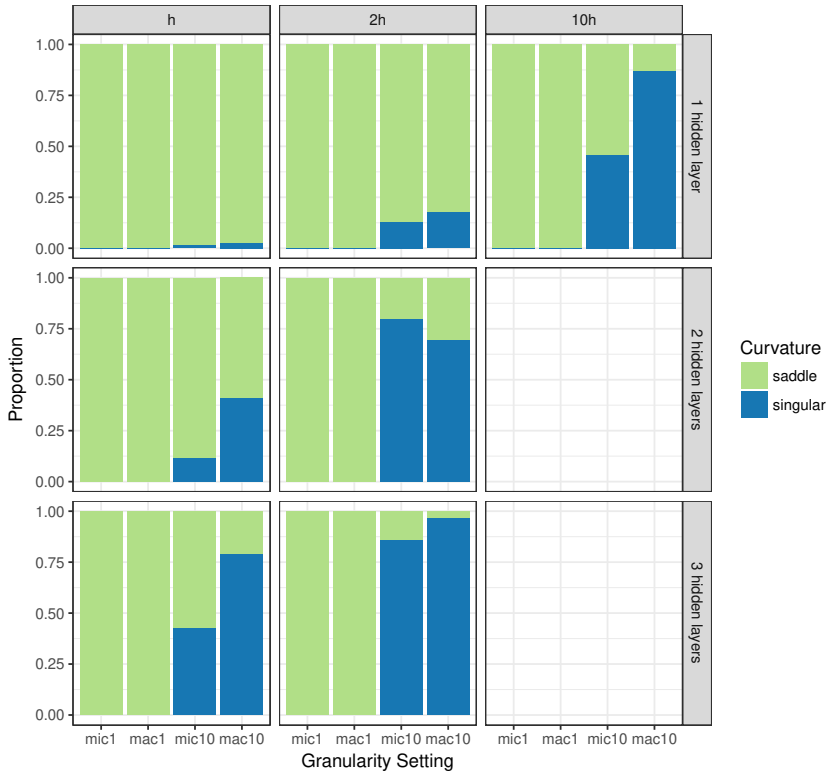
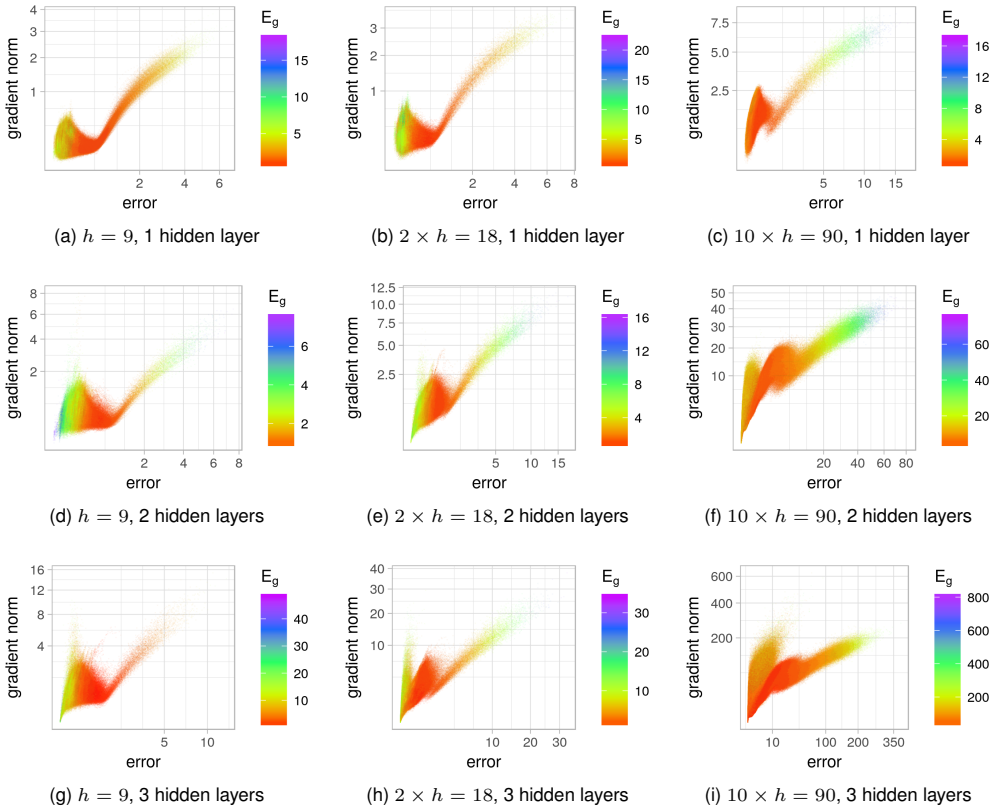


Figure 11: Histogram representation of the curvature information sampled by the gradient walks for the Glass problem for various NN architecture settings.



(a) $h = 9$, 1 hidden layer (b) $2 \times h = 18$, 1 hidden layer (c) $10 \times h = 90$, 1 hidden layer

(d) $h = 9$, 2 hidden layers (e) $2 \times h = 18$, 2 hidden layers (f) $10 \times h = 90$, 2 hidden layers

(g) $h = 9$, 3 hidden layers (h) $2 \times h = 18$, 3 hidden layers (i) $10 \times h = 90$, 3 hidden layers

Figure 12: L-g clouds colourised according to the corresponding E_g values, obtained by the $[-1, 1]$ micro walks for the Glass problem for the various NN architectures. An increase in the hidden layer size widened the attraction basin, and increased the steepness of the gradients. The cluster structure appears smoother and more connected for wider hidden layers. An increase in the number of hidden layers exaggerated the steep gradient cluster for all hidden layer sizes.

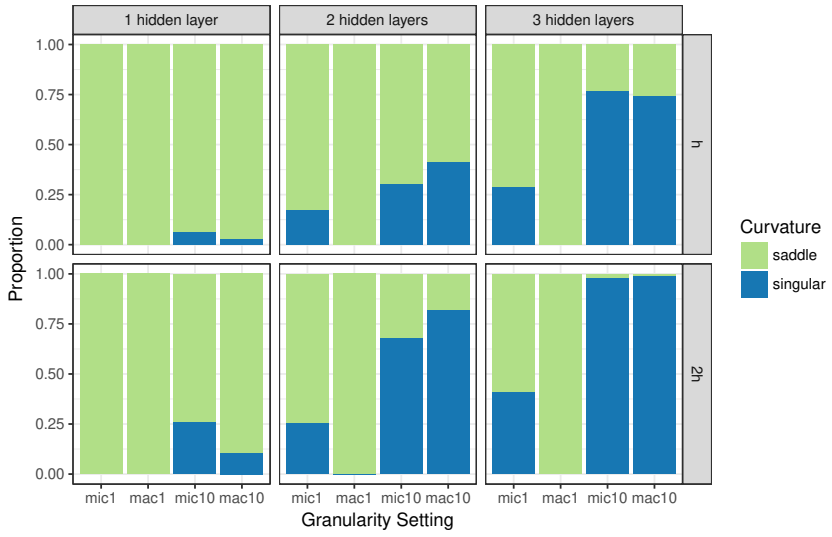
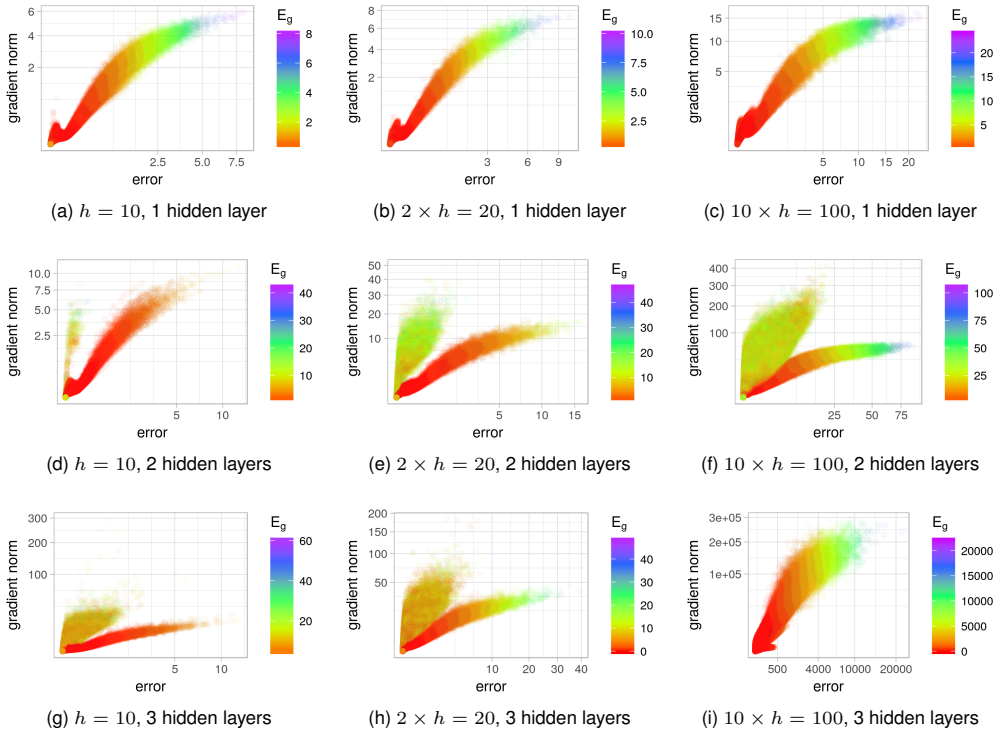


Figure 13: Histogram representation of the curvature information sampled by the gradient walks for the Cancer problem for various NN architectures.

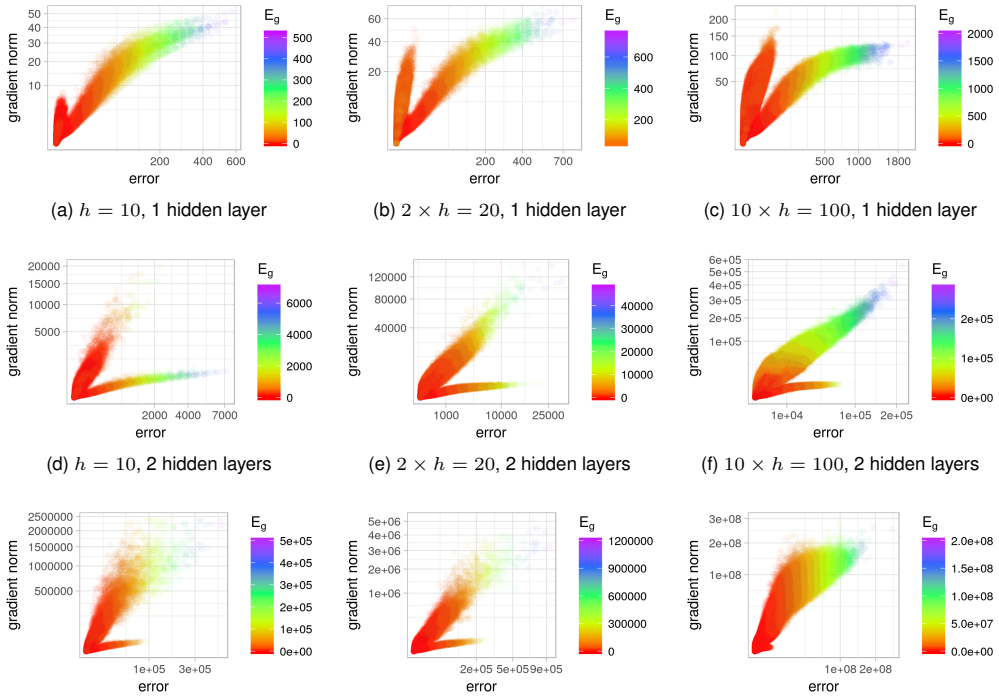


(a) $h = 10$, 1 hidden layer (b) $2 \times h = 20$, 1 hidden layer (c) $10 \times h = 100$, 1 hidden layer

(d) $h = 10$, 2 hidden layers (e) $2 \times h = 20$, 2 hidden layers (f) $10 \times h = 100$, 2 hidden layers

(g) $h = 10$, 3 hidden layers (h) $2 \times h = 20$, 3 hidden layers (i) $10 \times h = 100$, 3 hidden layers

Figure 14: L-g clouds colourised according to the corresponding E_g values, obtained by the $[-1, 1]$ micro walks for the Cancer problem for the various NN architectures. The overlap between the two clusters increased as the hidden layer width increased. Deeper architectures reduced the shallow gradient cluster and increased the steep gradient cluster.



(a) $h = 10$, 1 hidden layer (b) $2 \times h = 20$, 1 hidden layer (c) $10 \times h = 100$, 1 hidden layer

(d) $h = 10$, 2 hidden layers (e) $2 \times h = 20$, 2 hidden layers (f) $10 \times h = 100$, 2 hidden layers

(g) $h = 10$, 3 hidden layers (h) $2 \times h = 20$, 3 hidden layers (i) $10 \times h = 100$, 3 hidden layers

Figure 15: L-g clouds colourised according to the corresponding E_g values, obtained by the $[-10, 10]$ micro walks for the Cancer problem for the various NN architectures. NN depth and width exaggerate the steep gradient cluster.

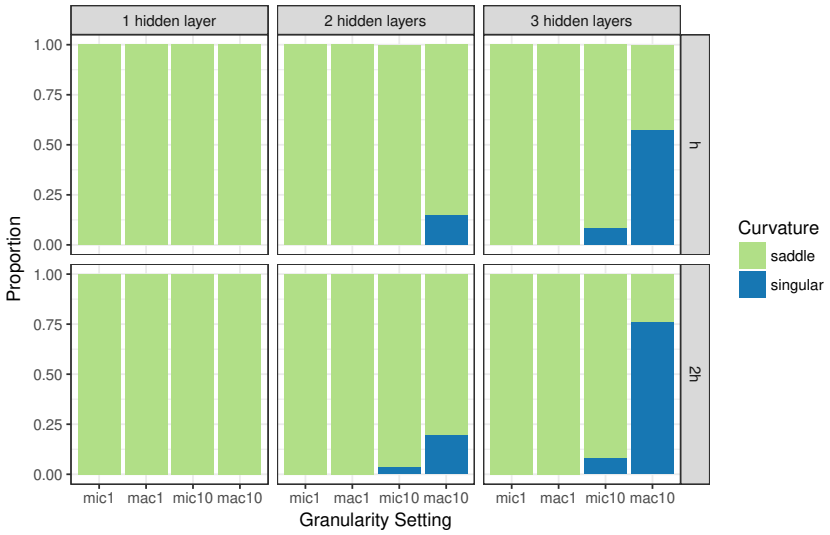
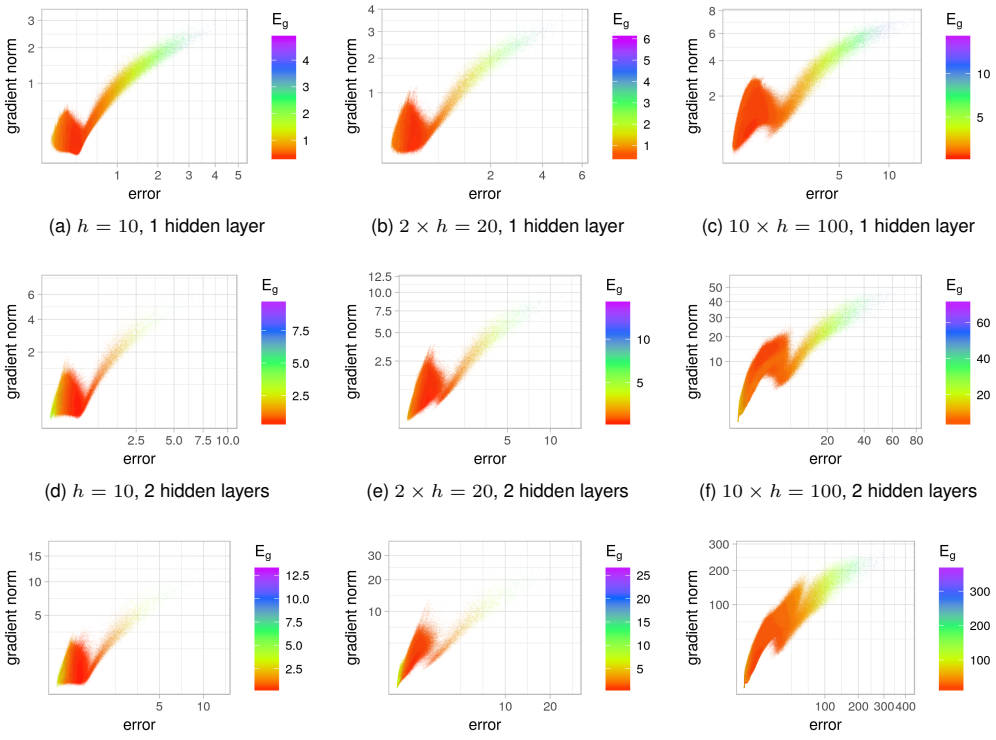


Figure 16: Histogram representation of the curvature information sampled by the gradient walks for the Heart problem for various NN architectures.



(a) $h = 10$, 1 hidden layer (b) $2 \times h = 20$, 1 hidden layer (c) $10 \times h = 100$, 1 hidden layer

(d) $h = 10$, 2 hidden layers (e) $2 \times h = 20$, 2 hidden layers (f) $10 \times h = 100$, 2 hidden layers

(g) $h = 10$, 3 hidden layers (h) $2 \times h = 20$, 3 hidden layers (i) $10 \times h = 100$, 3 hidden layers

Figure 17: L-g clouds colourised according to the corresponding E_g values, obtained by the $[-1, 1]$ micro walks for the Heart problem for the various NN architectures. An increase of the hidden layer width caused the non-global stationary attractor to become non-stationary. An increase in the architecture depth increased the overlap between the steep and shallow gradient clusters.

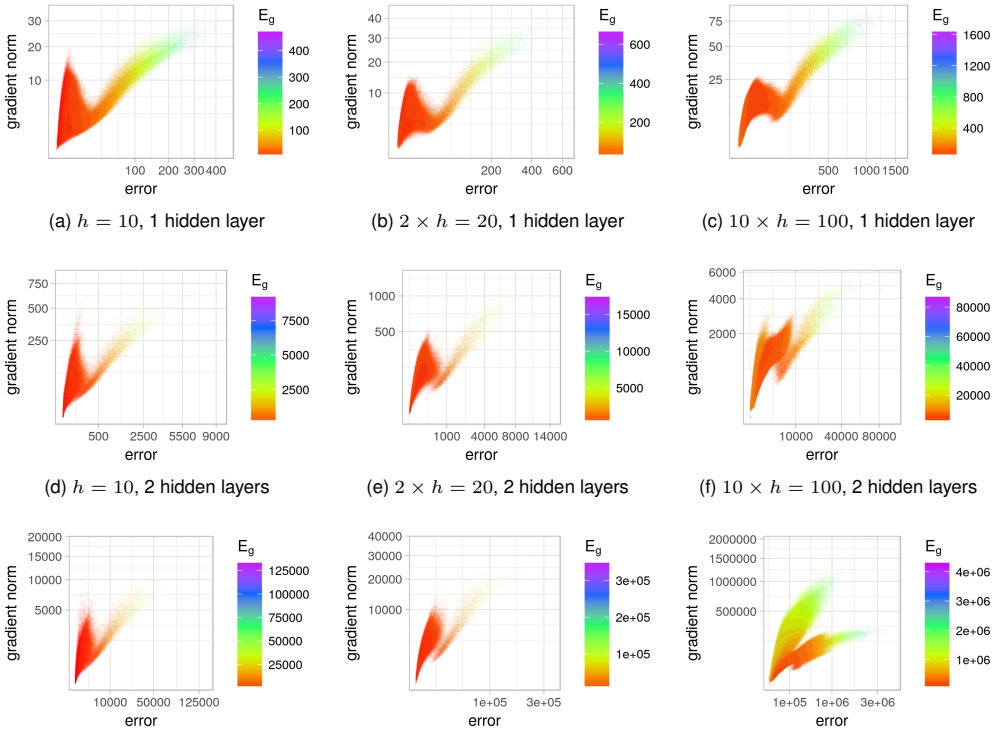


Figure 18: L-g clouds colourised according to the corresponding E_g values, obtained by the $[-10, 10]$ micro walks for the Heart problem for the various NN architectures. The width and steepness of the attraction basin increased rapidly as more hidden layers were added.