NEURAL LEARNING OF SPIRAL STRUCTURES

Sameer Singh, Department of Computer Science, University of Exeter, UK

ABSTRACT
Spiral structures are one of the most difficult patterns to classify. In this paper, some important characteristics of the two-spiral problem are discussed. The paper discusses the reasons why linear and non-linear approaches have difficulties with classifying such data. The paper focusses on how structural information about spirals can be useful in providing critical information to a neural network for their recognition. Results are presented on neural network solutions to the classical two-spiral problem by extracting structural and rotational information from the spiral training data.

1. SPIRAL STRUCTURES
Spiral data is found in several natural and physical domains. The classic double helix DNA, the motion of particles in cyclotrons, and spiral feed in manufacturing are some of the well-known examples. Spirals are particularly intriguing because of their high levels of non-linearity and resistance to shape transformation under rotation, translation or other scalar operations. Spirals structures are also attractive for their temporal properties and are found to be particularly hard to classify. For pattern recognition purposes, spiral recognition problems are specially attractive since we can manipulate their complexity with relative ease and control its size.

The spiral problem is a classic example of non-linear data. It is impossible to separate two spirals coiling around each other with a linear method. The spiral data is considered here in two dimension since previous work exists in this area for comparison. However, spirals can be generated in any number of dimensions. The benchmark spiral program, available from the Carnegie Mellon AI repository, generates two sets of points, each set with 96.density + 1 data points (3 revolutions of 32 times the density plus one end point). If a total of N data points are to be generated, then the spiral shape parameters change as follows, $1 \leq i \leq N$:

angle = ($i.\pi$) / (16.density) ...(1)
radius = maxRadius.((104.density) - i) / (104.density)) ...(2)
x = radius.cos(angle) ...(3)
y = radius.sin(angle) ...(4)

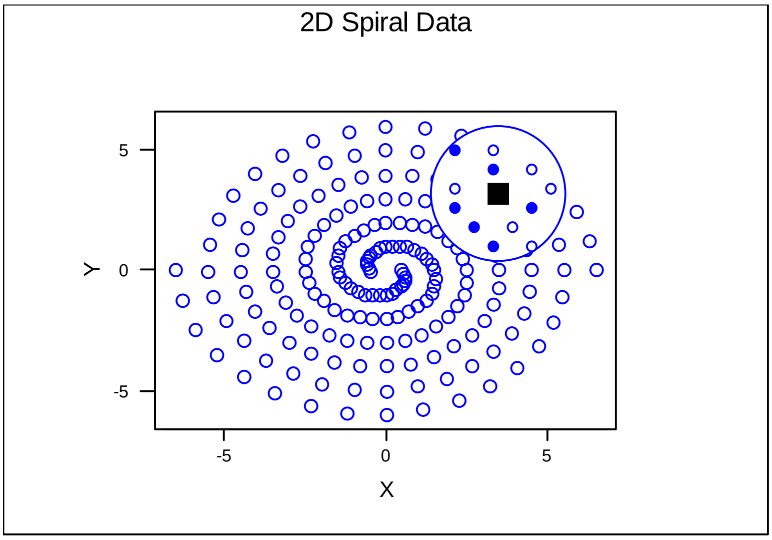
Here x and y are the spiral data points generated by the program, and π= 3.14. Since data points are generated in sequence, equations 1-4 are time dependent. The temporal nature of the resultant spiral is shown in Figure 1. Here the angle and radius of the spiral changes as new data is generated in sequence. The spiral is temporal because at a given point in time, the angle of the spiral that determines its position (x, y) is dependent on time (i in equation 1).

The two spirals are governed by three parameters: density φ, radius σ, and offset δ[1]. The density variable defines the total number of points generated within an envelope defined by

[1] The original spiral was proposed with $\varphi = 1$, $\sigma = 6.5$ and $\delta = .1$ (ref: Carnegie Mellon AI Repository)

the radius. Data belonging to two different classes lie on these two different spirals (represented as a sequence of white and black circles in Figure 1). By manipulating spiral parameters, it is possible to generate different spirals with varying radius and length.

Figure 1. 2D Spiral data scatterplot. Two spirals with a maximum radius of 6.5 coil around each other. The two different classes are highlighted in a hypersphere with their training data (white and black points) and a test pattern is illustrated with a black square.



A pattern classifier working on the problem should be able to recognise the training set before making test classifications. The benchmark has a training set, and test sets. The training set is represented by the vector {**x**, **y**}. The test sets may be thought of as noisy spirals, i.e. training set plus a uniform level of noise δ. For the two spiral problem, three test sets are possible: {**x**, **y**+δ}, {**x**+δ, **y**} and {**x**+δ, **y**+δ} where δ is a pre-defined scalar representing noise or offset. This method of offsetting test sets on different variables has been used for historical reasons than personal preference. It is possible to generate several different test sets of noisy spirals with varying δ. Uncertainty in pattern recognition occurs when: the offset δ is large for a small radius σ spiral or when data is dense, i.e. φ is large.

2. PREVIOUS RESEARCH

Recognising the two spiral benchmark is a difficult task for several pattern recognition approaches since spiral data is highly non-linear. The problem has been difficult to solve using neural approaches (see Touretzky and Pomerleau[1] for a discussion). It has been observed that backpropagation and its relatives encounter significant problems when training the neural network. In particular, deriving the optimal architecture is difficult, and furthermore, the training times are large. In addition, the spiral is under-constrained, i.e. data not lying on the spiral is often misclassified. For this reason, the two spiral problem has been particularly popular for testing novel neural and statistical pattern recognition classifiers. Considerable work has been done in the area since mid 1980s and in 1990s and a number of intelligent approaches have been applied to solving the spiral problem; neural networks:

Singh, S. "Neural Learning of Spiral Structures", Proc. International Conference on Advances in Pattern Recognition (ICAPR'98), Plymouth, UK, Springer, pp. 226-231 (23-25 November, 1998)

Fahlman[2], Fahlman and Lebiere[3], Lang and Witbrock[4], Tay and Evans[5]; neurofuzzy methods: Sun and Jang[6]; and data encoding methods: Chua et al.[7], Jia and Chua[8]. In addition, several other studies have tested their proposed pattern recognition methods on this benchmark problem since this process served as an indicator of their success with real-world problems, e.g. Ulgen et al.'s[9] hypercube separation algorithm's initial success with this benchmark confirmed superior results with hand-written character recognition data. Singh[10] used a single nearest neighbour method to recognise the two spiral data. Singh[11] used a fuzzy classifier to recognise spiral data and Singh[12] studied the effect of noise contamination of various types on the recognition performance. Singh[13] have also extended the fuzzy approach to recognising spirals in three dimensions.

3. LEARNING SPIRALS

The two-spiral problem, and its variants in higher dimension have some very interesting structural and spatial properties. If we are to study each spiral individually (*n* spirals in *n* dimension), we find that spirals are generated from the origin and move outwards in a helical structure. As they move outwards, the distance of the spiral from the origin increases and the distances between two successive points on the spiral also increase. However, these two distances are of little help for classification since they remain the same for both spirals and are not discriminatory.

The spirals have very interesting spatial properties. The overall structure of the spiral remains a spiral under rotation, translation or displacement. One of the methods for solving the problem may be the spiral unfolding such that the two spirals become linearly separable. However, this proposition is more difficult that it seems. The only other method of learning about spirals is to learn functions generating them using a black box type method. Neural networks, often claimed as universal approximators, have found spiral learning difficult. This is for two reasons. Spiral function is non-linear and time-dependent. Neural networks using standard MLP with backpropagation do not have the necessary information on the time-element through feedback loops to learn such a structure. Another reason is that both spirals, candidates in the discrimination process, have nearly the same function, except for some constant displacement scalar that preserves the distance between them. So, what additional tools do we have to enable ordinary neural networks to recognise different spirals?

Figure 2 shows the change in angle of the two spirals as it grows. Figure 3 shows how we may use this information. The angle θ is calculated as the change from (x_i, y_i) to (x_{i+1}, y_{i+1}) on the same spiral, rather than the angle of a given point from the origin with respect to the x axis. A close observation of the two spirals in Figure 1, and Figure 2 will reveal the fact that this change in θ for the two spirals at any given time is different from each other. This is a very important observation. If we are to calculate the sine of θ, then we find that this additional information gives us the power to discriminate between the two spirals.

The procedure now is to divide the training data into two halves; one for the first spiral and two for the second spiral. For each spiral, we compute the change in angle starting from the first point; if we represent spiral points as (x_i, y_i), then we create a new training file for each spiral with the information $(x_i, y_i, \sin\theta)$, where $\sin\theta$ is given by:

$$\sin\theta = (y_i - y_{i+1}) / \sqrt{(y_i - y_{i+1})^2 + (x_i - x_{i+1})^2}$$

The two new training files are now combined by interleaving the patterns from the two spirals. This training file is now fed to the neural network for training.

Figure 2. The two-spiral problem in three dimension. The spirals are separated on the basis of their class (left side of the class axis represents spirals or type1, and the right hand side represents spirals of type 2).

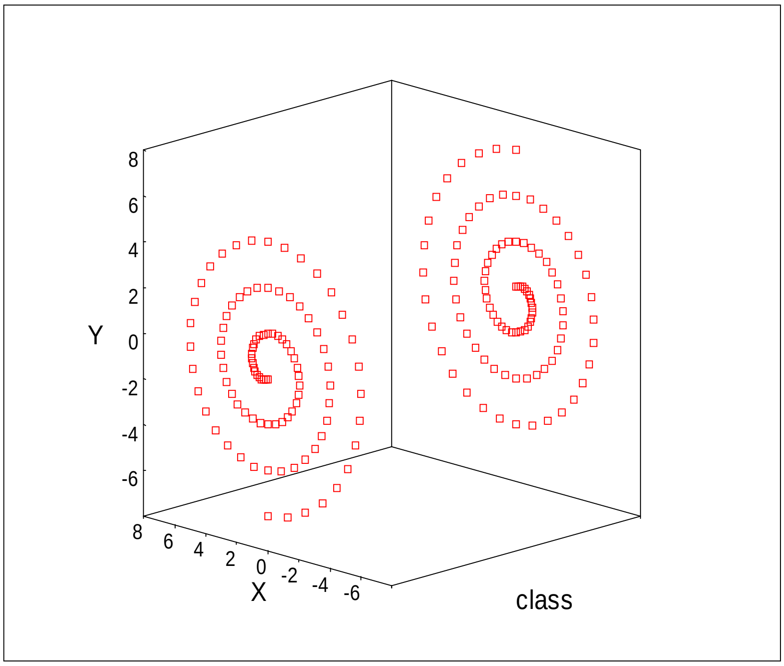
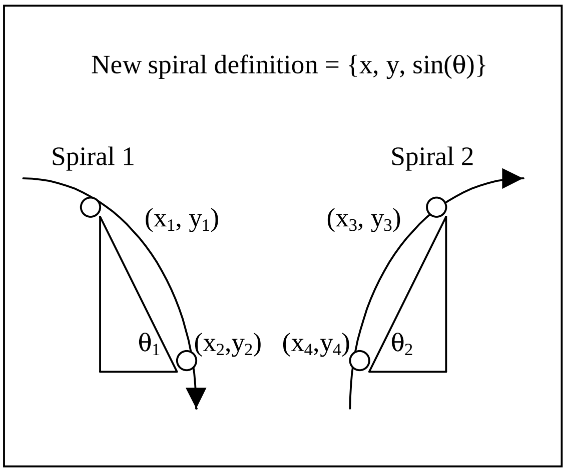


Figure 3. Change in angle with spiral movement

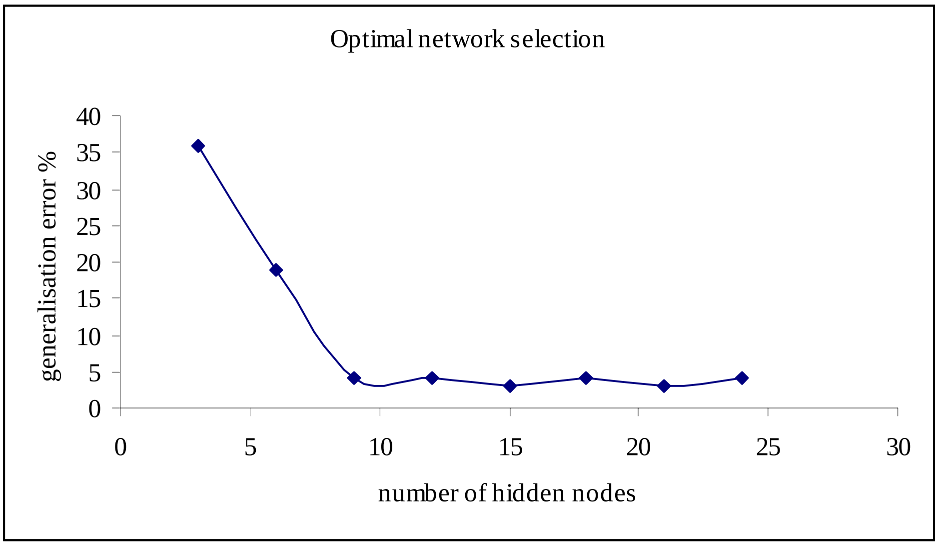


4. EXPERIMENT

The training data now consists of the tuples {**x**, **y**, **sin(θ)**}for spirals. The test data originally formulated as {**x**, **y**+δ}, {**x**+δ, **y**} and {**x**+δ, **y**+δ} is also represented as {**x**, **y**+δ, **sin(θ)** }, {**x**+δ, **y**, **sin(θ)** } and {**x**+δ, **y**+δ, **sin(θ)**}. For the test set, the sine of θ is calculated from the test data.

A simple multi-layer perceptron network using backpropagation with momentum is implemented with a learning rate of η=.1, and momentum of μ=.9. The network has the configuration 3x*h*x1 where *h* is the number of hidden nodes in a single hidden layer. The neural network is randomly initialised before the start of the training. The number of hidden nodes is increased in every trial and the generalisation error is noted. The number of hidden nodes is selected for the model that provides the least complexity and the least generalisation error. This is shown in Figure 3.

Figure 4. The selection of optimal network architecture



In Figure 4, the generalisation error decreases as the number of hidden nodes is increased. The increase in hidden nodes lead to a larger network that has a better generalisation ability. This performance however saturates after *h*=15. From Figure 4, we select the optimal number of hidden nodes equal to 15. Our experimentation is conducted from now on with a neural network having an architecture of 3x15x1. The three inputs to the network are the coordinates and the sine of the angle. The one output of the network is the class of the spiral, coded as 0 or 1 for spiral type 1 and 2.

5. RESULTS

The neural network was trained with an architecture of 3x15x1 for a total of 5000 epochs. The training and testing was performed using the Stuttgart Neural Network Simulator (SNNS). The training was performed with patterns from different spirals interleaved for a stable change in network weights. The trained network was then given the test data. The test data is produced for the three cases of spiral displacement as described before. For each case, the offset δ is varied from .1 to 1.0; these values are chosen as appropriate since for the spiral envelope of 6.5, each spiral coils roughly three times within this envelope as shown in Figure 1.

The test results are shown in Table 1. The recognition rate on different types of test sets is shown. Considering the combination of test data type and the offset δ, a total of 30 test sets have been used. The test results are very impressive. The neural network performance on classification approaches up to 97% correct recognition, and even with high values of spiral displacement in either one or both dimensions, the recognition rates remain particularly high.

Table 1. Testing on displaced data sets; spiral radius = 6.5

δ	Recognition rate % on $(x+\delta, y)$	Recognition rate % on $(x, y+\delta)$	Recognition rate % on $(x+\delta, y+\delta)$
.1	97	95	95
.2	97	92	92
.3	97	91	92
.4	96	92	91
.5	95	92	92
.6	94	91	91
.7	93	90	87
.8	92	88	86
.9	89	90	86
1.0	89	91	87

The results show that the neural learning of spiral structures is assisted by including training information on change in spiral angle with the generation of each successive point on the spiral.

6. CONCLUSION

In this paper, spiral structures have been recognised using the standard neural network method. The paper discusses why traditionally neural networks find it difficult to learn spiral structures without information on their temporal movement. When this information is available, neural networks are very efficient in their recognition. This work will prompt further work on using the same technique for recognising spirals in higher dimension.

REFERENCES

[1] Touretzky, DS, and Pomerleau, DA. What's hidden in the hidden layers? Byte 1989; August issue:227-233.

[2] Fahlman, SE. Faster-learning variations on back-propagation: An empirical study. In Proceedings of the 1988 Connectionist Models Summer School, Morgan Kaufmann, 1988.

[3] Fahlman, SE and Lebiere, C. The cascade-correlation learning architecture, In Advances in neural information processing systems 2, Touretzky, DS (ed.), Morgan Kaufmann, 1990.

[4] Lang, KJ and Witbrock, MJ. Learning to tell two spirals apart, In Proceedings of the 1988 Connectionist Models Summer School, Morgan Kaufmann, 1988.

[5] Tay, LP and Evans, DJ. Fast learning artificial neural network (FLANN II) using the nearest neighbour recall. Neural, Parallel and Scientific Computations 1994; 2(1):17-27.

[6] Sun, CT and Jang, JS. A neuro-fuzzy classifier and its applications, In Proceedings of the IEEE International conference on fuzzy systems, 1993, vol. 1, pp. 94-98.
[7] Chua, H, Jia, J, Chen, L and Gong, Y. Solving the two-spiral problem through input data encoding, Electronics letters 1995; 31(10):813-14.
[8] Jia, J and Chua, H. Solving two-spiral problem through input data representation, In Proceedings of the IEEE International conference on neural networks, 1995, vol. 1, pp. 132-135.
[9] Ulgen, F, Akamatsu, N and Iwasa, T. The hypercube separation algorithm: a fast and efficient algorithm for on-line handwritten character recognition, Applied Intelligence 1996; 6(2):101-116.
[10] Singh, S. A Single Nearest Neighbour Fuzzy Approach for Pattern Recognition, (in press, International Journal of Pattern Recognition and Artificial Intelligence, 1998).
[11] Singh, S. 2D spiral recognition using possibilistic measures, Pattern Recognition Letters 1998; 19(2):141-147.
[12] Singh, S. Effect of noise on generalisation in Massively Parallel Fuzzy Systems, (in press, Pattern Recognition, 1998).
[13] Singh, S. Massively Parallel Fuzzy Systems: The case of the three spiral recognition, In Proceedings of the IEEE International Conference on Fuzzy Systems FUZZ-IEEE'98, IEEE Press, 1998, vol. 2, pp. 1066-1071.