ANZIAM J. 45 (E) ppC935–C949, 2004

Chaos, fractals and machine learning

Robert A. Pearson*

(received 8 August 2003; revised 5 January 2004)

Abstract

The accuracy of learning a function is determined both by the underlying process that generates the sample as well as the function itself. The Lorenz butterfly, a simple weather analogy, is an example dynamical systems. Slightly more complex 6, 9 and 12, dimensional systems are also used to generate the independent variables. The non uniformly fractal distributions which are the intersection of the trajectories on a hyperplane are also used to generate variable values. As comparisons uniformly distributed (pseudo) random numbers are used as values of the independent variables. A number of functions on these hypercubes, and hyper-surfaces are defined. When the function is sampled near regions of interest and where the test set is of the same form as the learning set, both the chaotic system and fractal points have more accurate learners than the uniformly distributed

*School of Information Technology and Electrical Engineering UNSW, ADFA Campus, Canberra, ACT. Australia. Also 8 Sculptor St., Giralang, ACT. Australia. mailto:rpearson@netspeed.com.au

See http://anziamj.austms.org.au/V45/CTAC2003/Pea1/home.html for this article, Published August 28, 2004. ISSN 1446-8735

ones. Using one form of distribution to learn the data, and another for testing can be particularly poor. These cross distributional results are dependent of the functional form. Aspects of machine learning relevant to fractal distributions and chaotic phenomena are developed.

Contents

1 Introduction

A wide range of natural phenomena are complex dynamical systems that show some chaotic behaviour. The underlying chaotic system can determine other variables. The weather, which is chaotic, can impact on the economy. For example, farm incomes and price of produce to the consumer varies according to crop success or failure. The prediction of the interesting variable is complicated by the underlying chaotic system. Even with this complication machine learning techniques have been used to allow some forecasts [4]. Data mining of available sets is also more complex as data values are likely to be related to recent history as well as the current status of the system. Thus a general function learner is not sufficient. Further complications arise if all the variables of the underlying chaotic system are not available.

While a long term forecast is inaccurate, and accurate estimates of the current values are complex, some machine learning techniques can be useful. This paper uses some very simple chaotic models where data on all the independent variables are available. The variable to be estimated is known to be a function of the observed data.

2 Test dynamical systems

A chaotic dynamical system which is determined by an ordinary differential equation requires at least three variables. One such three variable system is known as the Lorenz Butterfly. This is derived from a set of partial differential equations that is a simple model of the weather. The ordinary differential equations for this are

$$\dot{x}_1 = \sigma(x_2 - x_1)\,, \tag{1}$$
$$\dot{x}_2 = \rho x_1 - x_2 - x_1 x_3\,, \tag{2}$$
$$\dot{x}_3 = \beta(x_1 x_2 - x_3)\,. \tag{3}$$

There are three critical points. One is at the origin. For the other two, $z_c = \rho - 1$ with $x_1 = \sqrt{\rho - 1}$, $x_2 = -\sqrt{\rho - 1}$, $y_1 = x_1$ and $y_2 = x_2$. For this data set $\sigma = 10$, $\rho = 28$, $\beta = 2.67$. The orbit of a realisation of this dynamical system tends to move into the neighbourhoods of a critical point then away again. It sometimes moves around an individual critical point, sometimes between them. As this is a chaotic system small changes in values can lead to large changes in the position of a solution.

Higher dimensional chaotic systems were generated using multiple linked three dimensional systems. Data for six, nine, and twelve dimensions were considered. The multiple sets were weakly nonlinearly coupled. This was cyclic, with the first being linked to the second, then the last back to the first. The second equation of each set was linked to the first of another by

$$\dot{x}_2 = \rho(1.0 + 0.02x_4)x_1 - x_2 - x_1x_3 \,. \tag{4}$$

The higher dimensional systems have multiple critical points. A single critical point consists of a selection of one of each from the sets. Thus the fourth dimensional system, including the ones at the origin has 3^4 points. Of these, 2^4 points are of particular interest as the regions near those with a zero value are not well populated with trajectories. To first order, the values of the critical points were the same as those of the three dimensional system. More accurate estimates of a selected point can be found using an iterative evaluation.

One functional value was calculated by averaging the values for similar equations. For example with four linked system $z = (x_3 + x_6 + x_9 + x_{12})/4$. Next the distance between this three dimensional value and the two three dimensional values of the first estimate of the critical points. A function of the three variables was defined so that the maximum was at one critical point, the minimum at another. Let r_1 be the distance from (x_1, y_1, z_c), r_2 be the distance from the point (x_2, y_2, z_c). Thus $r_i =$

$\sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_c)^2}$. The functional form chosen was

$$g(r_1, r_2) = 10\left(\frac{r_1 - r_2}{r_1 + r_2}\right).$$

To these values two types of noise were added. For the first the predicted value was multiplied by a uniformly distributed (pseudo) random values with a minimum of 0.99 and a maximum value of 1.01. Next the fourth decimal point of the recorded value of the coordinate was altered, again with a pseudo random number.

The function could be considered as a very simple analogue to the impact of the weather on the economy. If the nonlinear oscillator represents a very simplified southern oscillation (el Nino, la Nina) then the function measures the economic consequences of this on areas in Australia.

Another function used the distances between the coordinate value and the first estimate of critical points in the number of dimensions N used in the calculations. Thus for the distance to the first critical point with coordinates denoted by x_j^{crit1}, $D_1 = \sqrt{\sum_{j=1}^{j=N} x_j - x_j^{\text{crit1}}}$. The functional value was calculated from the distances to all the critical points with a non zero coordinate by

$$g = 10\left(\frac{\sum_{k=1}^{k=nc}(-1)^{k+1}D_k}{\sum_{k=1}^{k=nc}D_k}\right).$$

The critical points were calculated, and labelled by permuting the two options (+ve and −ve) with the lower dimensions varying the fastest. A third function was also considered.

The starting coordinates of the combined systems was $(0.001, 0.002, 0.003)$. Each successive system had (0.001) added to that of the previous one. Thus for the set of four ($x_{12} = 0.006$). The test set followed the trajectory after the learning set. If the same initial values on each set were used the trajectory is one where the values of each set remains synchronised. The small changes in the initial values of each set, together with the chaotic nature of each set, guarantees that coordinates cover a wide range of values. A second set of learning and test trajectories started at $(0.002, 0.003, 0.004)$ with (0.001) added for each group.

3 Fractal hyper-planes

The trajectory is a continuous curve in the hyper-rectangle. As discussed in [5] this curve intersects a (hyper) plane in a distribution that is essentially fractal. A fractal distribution is characterised by its fractal dimension. In addition the fractal of the intersecting points for the chaotic system does not cover the hyper-plane. There are significant regions without a point of intersection.

The points at which all trajectories intersect the $(x_1 = 0)$ hyper-plane were found to within the next 0.05 time-step. The functions defined on this were similar to the functions for the full dimensional trajectory. For the first function, the averages for the first combined dimension were calculated omitting the first coordinate. For the second function, the distances from the critical point of the first set omitted the first coordinate, and other sets were unchanged.

The trajectories for each hyper-rectangle were used for the (hyper) surfaces. Between 2,000 and 3,000 points occurred in the learning sets. The test sets had about 1200 points.

4 Pseudo-random

None of the trajectories will cover the possible multi-dimensional space. As well as being space in each dimension there are finite bounds. The bounds for each set of the Lorenz Butterfly in all combinations were similar, with minimum $(-13, -18, 0)$ and maximum $(13, 18, 56)$. A pseudo-random number generator was used and transferred to the range of the independent variable. This gives a uniformly distributed set.[1] Once the coordinates of the point were generated the functions were calculated in a similar way to those of the dynamical system.[2]

5 Results

An implementation of boosted regression trees [3] was used to learn an approximation of the functions. The set of trees was then used to predict the values of the test set and evaluate the errors. The results documented in the tables are the ratios of the error of the regression tree to the error of the naive prediction. This naive prediction is the mean value of the learning set. The different ratios are included. The Least Squares (LS) error is the measure used in building the trees. The LAD error is an alternative [1]. The maximum error is the error in an example which differs most from the prediction. In the tables, the ratio is the ratio of the largest absolute difference of the tree to the largest absolute difference in the naive prediction. The worst example for the tree is in general not the worst example for the naive prediction.

---

[1] Some pseudo random number generators use fractal distributions but unlike the fractal distributions generated by the chaotic dynamical system they cover the range.

[2] The MATLAB code for the dynamical system was modified for the pseudo random set. Also only one series of pseudo random numbers was used. This single series included the points, the noise added to the value of the dependent variable, and the noise added to the independent variables.

Table 1: Reduction in error for selected points on trajectories

			Two Point Distances				Function 2			
	Learning	Test	No of	Error Ratio			No of	Error Ratio		
n	Set	Set	trees	LS	LAD	Max	trees	LS	LAD	Max
9	Chaos	Chaos	61	0.11	0.32	0.56	21	0.05	0.21	0.39
9	Random	Random	41	0.26	0.51	0.63	22	0.10	0.30	0.52
12	Chaos	Chaos	62	0.13	0.36	0.53	20	0.05	0.20	0.43
12	Random	Random	75	0.20	0.44	0.61	11	0.12	0.34	0.56
9	Chaos	Random	61	0.71	0.89	0.84	21	0.65	0.86	0.70
9	Random	Chaos	41	0.25	0.49	0.65	22	0.14	0.33	0.55
12	Chaos	Random	62	0.53	0.77	0.70	20	0.63	0.83	0.70
12	Random	Chaos	75	0.23	0.47	0.61	11	0.18	0.38	0.62
Second trajectory with testing on the first										
9	Chaos-2	Chaos-1	34	0.16	0.40	0.65				
12	Chaos-2	Chaos-1	46	0.16	0.39	0.56				

5.1 Selected points on trajectory

The selected results for the various dimensions are included in Table 1. The second trajectory for the chaotic systems was also tested on the test set from the first one. Learning on one and testing on another had results with a similar accuracy to learning and testing on parts of the same realisation. In all cases learning on a chaotic set and testing on a similar set is more accurate than learning on a uniformly distributed one and testing on another set with the same properties. When a trajectory is used to learn and a uniform pseudo random set is used to test the errors can be large. The third functional form had results qualitatively similar to the other two.

Table 2: Reduction in error on a fractal set

			Two Point Distances				Function 2			
	Learning	Test	No of	Error Ratio			No of	Error Ratio		
n	Set	Set	trees	LS	LAD	Max	trees	LS	LAD	Max
8	Chaos	Chaos	30	0.11	0.32	0.39	45	0.02	0.12	0.29
8	Random	Random	37	0.28	0.53	0.64	7	0.07	0.27	0.49
11	Chaos	Chaos	54	0.13	0.36	0.52	32	0.02	0.14	0.30
11	Random	Random	42	0.39	0.57	0.66	12	0.06	0.24	0.44
8	Chaos	Random	30	1.27	1.28	0.76	45	0.82	0.93	0.79
8	Random	Chaos	7	0.24	0.45	0.55	7	0.24	0.45	0.55
11	Chaos	Random	42	0.32	0.56	0.72	32	1.06	1.06	0.80
11	Random	Chaos	12	0.22	0.45	0.60	12	0.22	0.45	0.60

5.2 Fractal distributions

Using only a hyperplane of point values gives similar qualitative results to the trajectory (Table 2).

6 Data set properties and learning theory

6.1 Properties of the dynamical system and the fractals

When considering learning theory aspects the properties of the dynamical system, and the fractal distributions are used.

For the dynamical system only the trajectory of the system generates points for the learner. Normal to the trajectory these points have a fractal structure [5]. Also in general this fractal structure does not cover the space. For example consider a dynamical system which has an unstable critical point. A particular trajectory may start near that point, but it will move away and not return close to the critical value. The actual points of the strange attractors will not be found in realisations.[3] Also with a chaotic nature the trajectories do not include limit cycles, nor do stable critical points occur in the chaotic regime.

The points distributed on the hyper-surface will not cover that surface. There will be regions with a significant measure where no point on the trajectory occurs. There will also be regions were two points are relatively close. Note that the fractal dimension is smaller than the dimension of the real variables.

6.2 Nearby examples

For a uniformly distributed random sample one of the curses of high dimension data is that even with a large data set the probability of finding another example close to a given one is small. In an alternative view the distance from a given example which is needed to find another example with significant probability increases. In the dynamical system the trajectory is continuous. Another example close to a given one depends only on the sampling of the trajectory.

If an arbitrary point is chosen the possibility of finding an example close to this point also differ significantly between the chaotic or pseudo random cases. For the pseudo random examples all regions, or points, are equivalent. For the dynamical system some finite regions never have any examples.

For the non-covering fractal sets some regions have significant numbers of examples, some none. Even if Gaussian noise was to be added to the fractal

[3]Consider the possibility that the values of the coordinates could be precisely represented in the word length. If these values were selected then ideally the trajectory remains at that point. In practice noise generates a trajectory.

set there could be significant regions with very low (or zero) probability of finding an example.

7 Hypothesis spaces examined

The theoretical convergence and its properties depend on the hypothesis space. Conceptually the functions are defined on R^n and probability distribution is conceived of as a (continuous) function in the appropriate number of (continuous) real variables. The boosted regression trees examine all hyper-rectangles with boundaries half way between the points. The hypothesis space actually examined in this paper depends on the data. For a random sample with a finite machine the number of possible distinct points is fixed by the word length. These points are uniformly distributed across the space defined by the representation. Conceptually, if the number of points was to become uncountable, and the word length was also uncountable, the hypothesis space corresponds to the space of the independent variables of the function. The smallest possible *rectangle* would be a point with measure zero. The hypothesis space is continuous.

For the dynamical system points are only defined on the trajectory. In the limit of an uncountable word length, and an uncountable number of examples the hypothesis space would become continuous line (linked along the trajectory). However, some rectangles will still have a finite size (non zero measure) normal to the trajectory. For some data sets, such as the atmosphere and the stock market,[4] only one trajectory exists. Thus all examples are obtained from a single trajectory.

The hypothesis space examined for the fractal distribution (here on the hyperplane) is also interesting. The possible examples are not continuous.

[4] While multiple stock markets exist they are linked. In this sense the set of stock markets form a single very complex system.

Even in the conceptual limit (of a single realisation) the probability distribution is not continuous. The fractal set in these examples does not cover the space. In the limit the regression trees (uncountably deep, and uncountably many) would find rectangles of finite measure.

Conceptually an ensemble of possible trajectories could be used to define a probability distribution. Even in this case some regions of the hyper-rectangle have very small, or even zero probability. The example points would be clustered in regions that are fractally distributed.

8 Discussion

The previous brief discussion on learning theory and these data sets helps to understand the results. The lower errors of both the chaotic (hypercube) and the fractal plane are related to the distribution of the examples in the various cases. The predicted value of an example on the test set is determined by the hyper-rectangle determined by the trees. The value is then determined from the value in this hyper-rectangle. This is in turn determined from the average of all values of the learner in the same region. If the region is small, or the function does not change rapidly, the accuracy is good. An accurate prediction occurs if nearby samples occur in the learning set. For a chaotic or fractal distribution a close value is likely to exist. The accuracy is high. In contrast learning on a chaotic set and testing on a random one can yield very large errors. This is related to the differences in the distributions. When a randomly selected point is chosen it can be in regions ignored by the chaotic one. No close point in the fractal set may occur. If this point is in a region where the function has significant changes in value then the differences between the test value and that of the nearest point in the learning set will be large. It is important that the trajectory be sampled for a time sufficient for it to move into all regions where the function is expected to have significant variation. Otherwise the errors can be large.

The test sets generated are artificial simple analogs to the weather. They form a fractal distribution. Such fractal distributions occur in a number of natural domains [6]. In data gathered from such sets the function itself may not be known. The only information is that of the observed points. The expected accuracy of the prediction may depend on the region into which a new point falls. If a number of points in the learning set are close then the accuracy is high. If no points are in the region the anticipated accuracy is low. In contrast the anticipated error of any uniformly distributed test set, based on a uniformly distributed learning set, is independent of the point.[5]

This paper has used the values of the chaotic system (or the fractal distribution), as independent variables and the current function values. No history is included. Often predictions are needed. This is more complex and often involves historical data [2]. Some of the features of the more simple *now-cast* can be extended to this more complex case. In particular if the system moves into a region which is not well represented in the learning data and the function has significant variability in that region the errors are anticipated to be large.

This paper has used a boosted regression tree, and this technique, in a sense, adjusts to the distributions as well as the changes in the function values. This type of adjustment is not necessarily replicated in other techniques. Case based learners can also adjust.

9 Conclusion

The process that generates function values determines the distribution of points that are available to both the learner, and the testing set. These distributions, together with the variability in the function will significantly affect the errors in machine learning. If a fractal set occurs the errors in test-

[5]Provided sufficient points are chosen to cover the domain.

ing on a fractal set can be expected to be significantly less than a uniformly distributed set of examples. Similarly data generated from a trajectory of a chaotic dynamical system can have significantly lower errors than those generated at random. It is crucial that a learner based on chaotic or fractal data is not presented with uniformly distributed data.

Acknowledgment: Calculations were performed at UNSW ADFA.

References

[1] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth International, Belmont, California, 1984. C941

[2] Robert A. Pearson. How to gain?/lose? on the stock market—datamining the ASX. In Vishy Karri and Michael Negevitsky, editors, *AISAT'2000*, pages 237–242, Hobart Australia, December 2000. C947

[3] Robert A. Pearson. Retrieval of atmospheric temperature profiles from satellite measurements using boosted regression trees. In Vishy Karri and Michael Negevitsky, editors, *AISAT'2000*, pages 128–133, Hobart Australia, December 2000. C941

[4] L. A. Smith. Sane, phychic and physchotic neural networks. In *European Geophysical Society Newsletter: European Geophysical Society XXIV General Assembly*, Le Hague, April 1999. C937

[5] Leon O. Chua Thomas S. Parker. *Practical Numerical Algorithms for Chaotic Systems*. Springer-Verlag, New York, 1995. C940, C943

[6] Various. Session np1 scaling multifractals amd nonlinear variability in geophysics. In *European Geophysical Society Newsletter: European Geophysical Society XXIV General Assembly*, Le Hague, April 1999. C947