Stability analysis of time series forecasting with ART models

A. Bocharov, D. Chickering & D. Heckerman
Microsoft Research, Redmond, WA, USA

Abstract

Time Series (TS) analysis based on autoregressive tree models has been introduced in Meek et al. (2002). In addition to features present in the original design, the current SQL 2005 implementation also addressed the "forecasting instability" phenomenon that has been observed on a range of highly non-stationary and/or non-linear data sets.

In technical terms the cases of long range forecasting instability are characterized by rapid growth of the mean absolute prediction error with time, which may or may not be accompanied by significant growth of the predicted standard deviation. In practice, the cases of instability where predicted standard deviation stays tame are especially misleading, since they can furnish unreliable predictions with little or no visual cues that would characterize them as unreliable.

The method described in this paper is designed to detect and control the long range forecasting instabilities and to cull the unreliable predictions.
Keywords: time series, forecasting, autoregression, stability, data mining.

1 Autoregressive tree (ART) models

Consider historic time series data for multiple time series. For each target (predictable) series, we build *autoregressive tree model*, which is effectively a collection of autoregressive models [2] and where the boundaries between the applicable models are defined by binary decision trees (see subsection 2.1 of the research paper [1] for detailed definition and examples of ART models).

When underlying decision tree for a target time series Y consists of more than one leaf node, then each of its interior nodes is defined by an input variable V

WIT Transactions on Information and Communication Technologies, Vol 37, © 2006 WIT Press
www.witpress.com, ISSN 1743-3517 (on-line)
doi:10.2495/DATA060141

and a threshold h: subtrees below the interior node are defined by $V < h$ and $V > h$ or $V \geq h$ conditions. A section of such decision tree might look like this

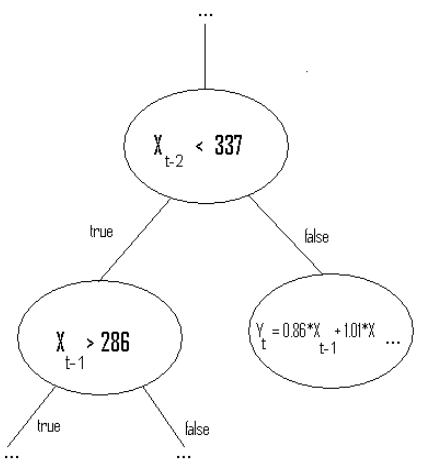

Figure 1: A section of an autoregressive tree.

We say that an interior node represents a *split* and we call the branching condition of the node a *split condition*. We also often refer to the variable and the threshold of that condition as *split variable* and *split threshold*. (E.g. the diagram above shows two interior nodes – one representing a split on X_{t-2} and the other representing a split on X_{t-1}).

The *depth* of an ART model is the maximum time lag appearing in all the constituent leaf AR models and in all the split conditions. Given an ART model of depth p, it suffices to have a history of depth p for all the input time series' in order to make one-step forecast for all the output time series.

Indeed, given history of that depth, every split condition in every decision tree resolves to either *true* or *false* and therefore we end up with a single prediction leaf, i.e. with a specific AR model of depth at most *p*, for each target series. We have enough input values to seed all the AR models we have selected and thus to obtain the predict distribution Y_{t+1} for each target time series *Y*.

The current version of ART analyzer uses the *time* index as an input variable in addition to time series inputs. Therefore, the ART decision trees often have splits on time in addition to splits on input series. So while the original design [1] addressed the non-linearity of data by approximating data with a set of piecewise-linear models, the current version also addresses the non-stationarity by making the models piecewise-stationary.

As per the autoregressions used, ART puts an emphasis on the quality of the first forecasting step (as would any other modeling technique based on classical

autoregressions). Indeed, an ART model optimizes the Bayesian score of a collection of distributions that have the form

$$p(Y_t \mid X_{1,t-1},...,X_{r,t-1},....,X_{1,t-p},...,X_{r,t-p})$$

The multi-step joint distributions such as

$$p(Y_{t+k},....,Y_t \mid X_{1,t-1},...,X_{r,t-1},....,X_{1,t-p},...,X_{r,t-p})$$

are not directly learned from data and have to be emulated by way of Monte-Carlo process called *forward sampling*, that is described in the next section.

The one-step forecasting with ART models has been found superior compared to other general-purpose AR methods (see the evaluation section in [1]). It turns out that multi-step forecasting using AR models has a complication, to which ART is not immune either, which is the sampling *instability* phenomenon, explored in detail in sections 2, 3, 4.

2 Long range forecasting based on forward sampling and the instability phenomenon

As explained in the previous section, ART structures model piece-wise stationary, piece-wise linear data well. When the data is strongly non-stationary, the trees tend to have "splits on time" in order to capture the change in statistical patterns of the data over time.

Let us assume that the most recent such time split occurred T_s time units in the past. In absence of seasonality lags only patterns recorded after that split will be used in forward prediction. For a long range forecast we commit to the assumption that the applicable past data patterns (such as patterns recorded up to T_s steps into the past) will persist into the future. Forecasting by forward sampling process described in this section does not introduce new splits and does not, in itself, have any other mechanisms for controlling inherent prediction non-stationarity.

Here is the outline of the forward sampling algorithm for forecasting. (This approach explored in [1] can be viewed as a "continuous" analog of discrete Bayesian sampling introduced in [3].)

Let $Y_1,...,Y_m$ be a set of time series' with histories available up to the moment of time t and let M be an ART model describing this set of series' based on the available history. Straight from this model we can generate distributions for predicted future values

$Y_1[t+1],...,Y_m[t+1]$. We cannot immediately do the same for the future time point $t+2$, since in general that would require knowledge of values $y_{1,t+1},...,y_{m,t+1}$ to be used as inputs into the model.

The process of forecasting over the range of time $[t+1,...,t+k]$ involves a sufficiently large number N of sampling iterations.

At each iteration we build a sequence of sampled vectors

$$\{y^*_{1,t+1},\ldots,y^*_{m,t+1}\}$$

.....

$$\{y^*_{1,t+i},\ldots,y^*_{m,t+i}\}$$

To build the first of these vectors we obtain the predict distributions $Y_1[t+1],\ldots,Y_m[t+1]$ from the model using the actual historical data as input, then draw $\{y^*_{1,t+1},\ldots,y^*_{m,t+1}\}$ as samples from these distributions. We can now extend the series' with these values to get $Y_1+\{y^*_{1,t+1}\},\ldots,Y_m+\{y^*_{m,t+1}\}$ curves to be used as inputs in further sampling steps.

In general, at step $i, 1 \le i < k$, assuming we have already generated the extended series $Y_1+\{y^*_{1,t+1},\ldots,y^*_{1,t+i}\},\ldots,Y_m+\{y^*_{m,t+1},\ldots,y^*_{m,t+i}\}$ we obtain predict distributions $Y_1[t+i+1],\ldots,Y_m[t+i+1]$ using these extended series as the input to the model, then we draw the new values $\{y^*_{1,t+i+1},\ldots,y^*_{m,t+i+1}\}$ as samples from these distributions and append them to the extended series'.

After performing N full sampling iterations we have N emulated future values for each series Y_r at each future point $t+i, 0 < i \le k$. Using these emulated future values we can compute estimated joint distributions for sequences of predicted values (for each $i \le k$).

Depending on the properties of the ART model and of the data it represents, the forecasting process as described above can be *asymptotically stable*, *practically stable* or *unstable*.

In the context of ART forward sampling these notions are similar to the notion of stability of solutions of stochastic differential equations [6, 7]. The principal question to ask is: what happens to the forecast when we apply a sufficiently small perturbation to the original time series data?

When any small data perturbation can cause forecasting difference that would persistently grow with time, we characterize the predictions as *unstable*.

In general, such predictions are unreliable, because, looking at a resulting volatile behavior we have no way of knowing whether the prediction volatility is an accurate reflection of the inherent volatility of data, or it is rather due to problems with the forecasting process.

2.1 Example

Shown below in Figure 2 is a synthetic curve, where an exponential growth pattern gives way to a flat line in the end. Despite the change in pattern there are stationary AR formulas with near-maximum likelihood that appear to model long range prediction well. Let us compare the behavior of the second-order AR formula

$$Y_t = 0.308 + 1.762 * Y_{t-1} - 0.779 * Y_{t-2}$$

and this third-order AR:

$$Y_t = 0.33 + 1.762 * Y_{t-1} - 0.54 * Y_{t-2} - 0.24 * Y_{t-3}$$

In the absence of noise, both models seem to commit nicely to the flat pattern in the end of the data. However when noise is added, to a tune of 2% of the signal, to the original data, the prediction curves deviate from the ideal to various extent. It is easy to emulate 20-step forward noisy prediction with either formula. When done with the 2nd order AR the forecast mean deviates from the ideal prediction by a maximum of 0.82 around the 10th step then starts converging with the difference shrinking to 0.4 by the 20th step. With 3rd order AR the difference grows to approximately 2.45 towards 15th step and then stays at that general level. It is clear that here the effect of noise is much more dramatic and the stability of the 3rd order AR is questionable.

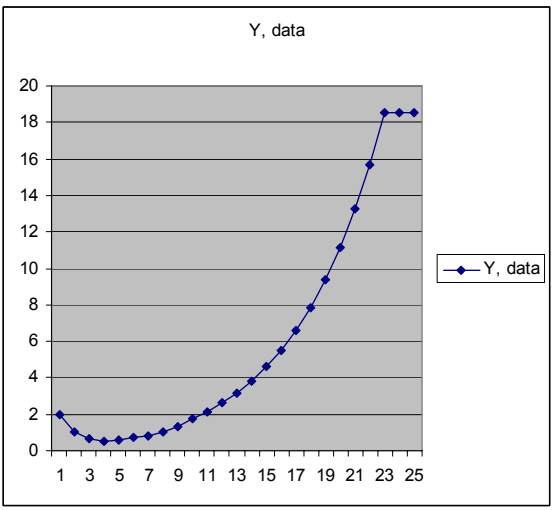

Figure 2: Noise-free data for Y.

In this patently simple example with one time series and no splits the stability analysis can be done analytically without running the actual predictions. But in really rich ART models the analytical method does not seem to be promising.

3 Forecasting stability analysis

The method we use for monitoring stability of long range forecasting is based on exploring the divergence rate of the forward-sampling operator.

Expanding on the description of forward sampling given in the previous section, let us introduce a shorthand notation Z_{t+i} for the whole subsequence of sample values

$$\{y^*_{1,t+1},\ldots,y^*_{m,t+1}\}$$

$$\{y^*_{1,t+i},\ldots,y^*_{m,t+i}\}$$

accumulated towards future time point $t+i$.

Let us view the next sample $Y^*_{t+i+1} = \{y^*_{1,t+i+1},...,y^*_{m,t+i+1}\}$ as a result of application of a forward sampling operator FS_i to Z_{t+i}

We need to explore the behavior of this hypothetical operator between different forward sampling iterations. Suppose a and b are labels of two such iterations, $Z_{a,t+i}$ and $Z_{b,t+i}$ are the respective forward sampling collections at the point of time $t+i$ and $Y^*_{a,t+i+1}$, $Y^*_{b,t+i+1}$ are the respective further samples.

Assume for the moment that $Z_{a,t+i}$ is fixed and $Z_{b,t+i}$ is being generated synthetically multiple times to be very close to $Z_{a,t+i}$. Then we could define

$$Norm(FS_i \mid Z_{a,t+i}) = \lim_{\varepsilon \to 0} \max(\frac{dist(Y^*_{a,t+i+1}, Y^*_{b,t+i+1})}{\varepsilon} \mid dist(Z_{a,t+i}, Z_{b,t+i}) = \varepsilon)$$

and

$$Norm(FS_i) = \max\ Norm(FS_i \mid Z_{a,t+i})$$

The norm, thus defined, measures the rate at which the operator FS_i can amplify the mispredictions accumulated by the point of time $t+i$.

If $Norm(FS_i) < 1$ then sufficiently small mispredictions are dampened, not amplified at the i-th step. As long as $Norm(FS_i) < 1$ we can have a stable forward sampling process for these values of i.

The norm, as defined above, is largely a theoretical measure, which would be not practical to compute in a real world computer application, should it even be algorithmically possible. So in practice we use empirical values called divergence rates, that are inspired by the norm but give a coarser measure of stability.

Consider the set $Iter \times Iter$ of pairs of forward sampling iterations. Let S be a large enough subset of $Iter \times Iter$ not containing any pairs of the form (a,a). (At maximum S could be the entire $Iter \times Iter \setminus diagonal$.) For each $i = 1,...,k$ then consider the mean inter-sample distance

$$d_{i,S} = (\frac{\sum_{(a,b) \in S} dist(Y^*_{a,t+i+1}, Y^*_{b,t+i+1})^2}{|S|})^{1/2}$$

Now define divergence rate at point $t+i$ to be $d_{i+1,S} / d_{i,S}$. This number gauges the expected growth of mean inter-sample distance from step to step. If this growth rate is significantly greater than 1, we should expect any misprediction at step $t+i$ to translate into a larger misprediction at step $t+i+1$. If the mean divergence rate stays significantly greater than 1 for sufficiently large number of consecutive steps, then it casts doubt on the stability, and reliability of the predictions.

The fine art of using the divergence rates depends on good answers to these three questions:

- What is a good support subset S in $Iter \times Iter$ to compute these numbers with?
- What does "significantly greater than 1" mean?
- What is "sufficiently large number of consecutive steps" after which we should pronounce the forward sampling process practically unstable?

The support subset S turns out to be a major tuning factor for the algorithm. The general idea is to select such pairs of iterations (a,b) where $Z_{a,t+i}$ and $Z_{b,t+i}$ are sufficiently close to each other. The closeness is a relative measure, of course, and a good scale against which it can be defined is provided by the covariances of predict distributions up to the time step $t+i$. For example, when Y consists of just one (scalar) time series, the geometric mean of standard deviations of predict distributions $Y_a[t+i]$ and $Y_b[t+i]$ provides a good measuring scale which means, in practice, that the correlation coefficient of these two distributions would be a good measure of closeness.

Let us address the second question.

First consider an ART model that does not have splits for any of the targets (effectively, a collection of AR models). For such, a divergence rate (d.r.) of greater than 1 at any point means that the quality of the prediction at the next step is likely to decline. It depends on the length of forecasting range and on the user quality requirements, whether this can be tolerated for a certain number of steps. In our experience, the d.r. >1.5 at the very first step meant in most cases that the model is going to be invalid for further forecasting. D.r. between 1 and 1.5 was more like a market alert – it made sense to try further forecasting steps in hope that the rate would eventually drop.

In case the model is a proper ART with a number of spits, the use of d.r. is more complicated at threshold points, however we skip the detailed discussion of it due to lack of space.

Now we can turn to the third question, namely: for how many steps should we tolerate a high divergence rate? Given the non-linear non-stationary character of rich ART models, the divergence rate can go up and down from one forecasting step to another. If a maximum overall error amplification rate $A_{\max}$ is set for practical purposes then it is prudent to stop the forecast at a step where the running product of divergence rates exceeds $A_{\max}$.

4 Real world examples

4.1 Mega millions lottery

Let us try and predict US Mega Millions lottery winning picks. To this end we can download the history of winning numbers from http://megamillions.com/winningpicks/download_numbers.asp from, say, January 2005. The lottery is the usual "guess 6" setup: 5 regular numbers and a "Mega Ball" are drawn from 1 through 55.

At certain parameter settings we get a TS model with the following subset of relevant leaves:
Mega Ball: (Unconditional): Mega Ball = 24.42, variance < 1.0
Pick1:(if Pick5(t-3) > 51 and Pick3(t-1) < 22)
Pick1 = -49.56+0.97*Pick5(t-3) +0.17*Pick3(t-1), variance 9.59
Pick2: (Unconditional) Pick2 = 18.23, variance < 1.0
Pick3: (if Pick5(t-3) > 44)
Pick3 = 29.62+0.02*Pick5(t-3), variance 132.95
Pick4: (if Date > 3/12/2005) Pick4 = 37.45, variance < 1.0
Pick5: (if Pick2(t-5) < 16 and Date > 6/19/2005)
Pick5 = 54 – 0.22*Pick5(t-5)+0.72*Pick2(t-5), variance 21.18

Since lottery picks are certifiably random, this model can not possibly work. So what kind of telling signs should we be looking for? Huge variance in the AR for Pick3 is one such sign, but an even stronger one is the model divergence rate (d.r.) of **1.88** at the first prediction step, which is way more than the most generous empirical cap of d.r.=1.5.

4.2 High-complexity "Economy Indicators" model

Consider a 10-year history of monthly key economic indices, such as a cleaned up snapshot of data available at http://www.economagic.com/popular.htm . In this experiment the list of indices included "Discount rate", "Federal Funds rate", "Single family housing starts", "Multiunit housing starts", "New construction total", "Industrial Production Index", "30 Year Fixed mortgage rate", "New computers production", "New defence orders", "New durable goods production", "New motor vehicles production", "New power production", "Per-capita income", "Bank prime loan rate", "Treasury bill interest rates", "Unemployment rate", "Yen to US dollar exchange rate" plus three dozen more of similar measures.

A parameter setting of the algorithm adjusts the complexity of autoregressive trees, which leads to initial ART model **M1**, that is high branched with the total of 156 leaves across the 55 target trees and on average 6.5 regressors per leaf. It turns out that such high complexity is not good for the forecasting stability. For the first 4 prediction steps the divergence rates amount to **1.54**, **2.66**, **9.17** and **19.99**, indicating an unreliable forecast.

Adjusting parameters of the method leads to a more robust model **M2** with about 105 leaves across 55 target trees that average 4.7 regressor per leaf. The first 4 divergence rates go down to 1.02, 1.175, 1.12 and 1.05, which spells moderate and practically acceptable error amplification potential. (See also a discussion of this data set in [4].)

Table 1 below presents an essential trade-off comparison between the two models using a held out a 6-point portion of each of the listed. The score columns contain the average prediction log-likelihoods computed using the holdout points. (At 7th step predictions with M1 were becoming severely impaired: e.g., predict means for "Discount Rate" were 2.77, 3.23, 2.13, 3.64,

3.07, -2.13, +20.94). The StdDev columns list the average predict standard deviation over the 6 predictions.

Table 1: Selected comparative uncertainties and scores for models M1 and M2.

Series	Scale	M1 StdDev	M2 StdDev	M1 score	M2 score
Discount rate	4.5	2.89	0.35	-5.9	-6.0
Federal rate	4.5	0.74	0.37	-6.27	-5.95
Housing single	1300	34	52	-6.82	-5.68
Housing multi	300	49.4	37.3	-4.98	-4.78
Industr.production	140	0.98	1.26	-1.42	-2.03
Inflation	4	2.6	2.4	-3.33	-3.4
Mortgage 30yr	6	0.41	0.43	-0.41	-0.24
New motor veh.	37000	1298	1695	-10.0	-9.2
New power	2000	425	430.5	-7.5	-8.15
Per-capita income	9000	42.4	40.7	-7.64	-5.47
Prime rate	7.5	0.67	0.38	-5.59	-4.74
3-month T-bill	4.5	0.44	0.4	-5.84	-5.1
6-month T-bill	4.5	0.42	0.51	-6.36	-3.86
Unemployment	5	0.167	0.187	-4.23	-6.57
Yen to dollar	120	5.54	7.23	-3.17	-3.32

The bottom line of this comparison is that the somewhat coarser model M2 has good short-range predictive quality comparable to that M1 (better score on 9 and worse on 6 of the above listed targets). M2 however has a huge advantage of producing stable long-range forecasts.

5 Conclusion and further research

The forecasting stability analysis method described in this paper is designed to flag unreliable predictions that can occur in AR and ART models due to instability of the forward-sampling process. Expanding this method to Time Series models of other types would be an important direction of further research, as it should improve our understanding of the value and usability of Time Series forecasts.

References

[1] C. Meek, D. Chickering and D. Heckerman. Autoregressive tree models for time-series analysis. In *Proceedings of the Second International SIAM Conference on Data Mining*, Arlington, VA, SIAM, April, 2002

[2] J. Hamilton. Time Series Analysis. *Princeton University Press*, Princeton, NJ, 1994

[3] M.Henrion. Propagation of uncertainty by probabilistic logic sampling in Bayes' networks, in Kanal, L. and Lemmer,J. ,editors, *Uncertainty in Artificial Intelligence 2*, pages 149 – 164, North-Holland, New York, 1988
[4] A. Bocharov, J.Lind. Data Mining Reloaded. In *SQL Server Magazine*, February 2005 (InstantDoc #44281)
[5] Z.Tang, J. MacLennan. Data Mining with SQL Server 2005, Wiley, Indianapolis, IN, 2005
[6] D. Talay, Efficient numerical schemes for the approximation of expectations of functionals of the solution of an SDE and applications. *Springer Lecture Notes in Control and Inform. Sc. 61* 1984, pages 294-313.
[7] D. Talay, L. Tubaro, Expansion of the global error for numerical schemes solving stochastic differential equations. *Stochastic Anal. Appl.* ***8*** , 1990, pages 483-509.