Информационные процессы, Том 13, № 4, 2013, стр. 401–414.

Time series homogeneity tests via VLMC training

M. Malyutov, T. Zhang, Yi Li, and X. Li

Department of Mathematics, Math. Dept, Northeastern University, 360 Huntington Ave., Boston, MA 02115, USA
E-mail: m.malioutov@neu.edu, li.xin2@husky.neu.edu, zhang.tong@husky.neu.edu and li.yi3@husky.neu.edu
Received December, 10, 2013

Abstract—J. Rissanen modified Markov Chains of memory length n by assuming that the current state of a string is independent of the symbols in its past preceding a **context** of the length which is a function of the past itself. He developed a construction algorithm of contexts stochastic suffix tree for compressing *stationary* time series and proved its *consistency.*
Rissanen's Variable memory Length Markov Chains (VLMC) were formally applied for classifying non-stationary proteins by G. Bejerano and others by training their VLMC models.
In this paper, we apply VLMC training of presumably stationary parts of piece-wise stationary time series for testing homogeneity of longer regions and, if homogeneity is rejected, for identifying most discriminating contexts.

KEYWORDS: Variable memory Length Markov Chain, log-likelihood, approximate likelihood ratio homogeneity test, mining for most distinguishing contexts.

1. INTRODUCTION AND TESTING ALGORITHMS

Modeling random processes as full n-Markov Chains (MC) can be inadequate for small n, and over-parameterized for large n, e.g. if the cardinality of the base state space is four, $n = 10$, then the number of parameters is 3,145,728. The Box–Jenkins ARIMA approach in quality control, popular since the sixties, is not adequate in applications to linguistics, genomics and proteomics, security, etc, where comparatively long non-isotropic contexts are relevant that would require huge memory size of the full n-Markov Chain (MC).

The popularity of another tool – sparse Variable Memory Length MC (VLMC), has been increasing rapidly since its invention in [14] for compression and – in the 21st century, also for classification aims in genomics, proteomics, etc.

Sparse VLMC over some alphabet A is a very special case of n-MC, where n is the maximal length of **contexts**. With a given string $x_{-n}, \dots, x_{-1}, x_0$ and certain abuse of notation we define

$$C(x_0) = x_{-1}, \dots, x_{-k}, k \leq n := x_{-1}^{-k}, \tag{1}$$

(to a current state x_0) is a subsequence of the preceding states x_{-1}^{-n} (see (1)) of the **minimal length** such that the conditional probability

$$P(x_0|x_{-1}^{-m}) \equiv P(x_0|x_{-1}^{-k}), \forall m > k. \tag{2}$$

For large n, VLMC is sparse, if the total number of contexts $|\tau|$ is $O(n^k)$ for some fixed k. VLMC can be viewed as probability suffix tree, see Figure 1.

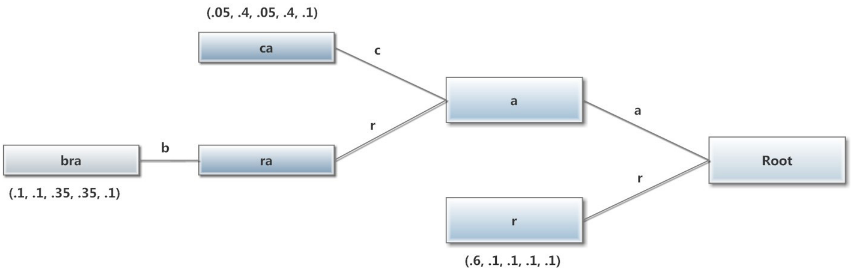

Figure 1. Illustrative example of stochastic context tree with distributions of the root given contexts written under the leaves of the tree

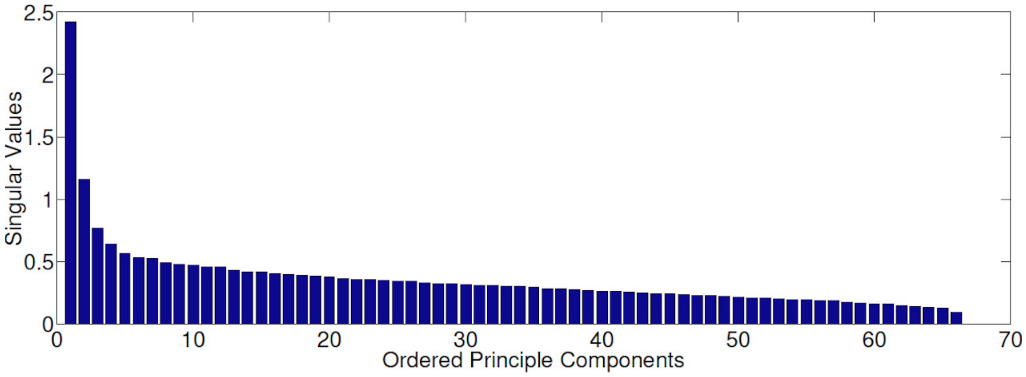

Figure 2. Singular values of daily log-returns

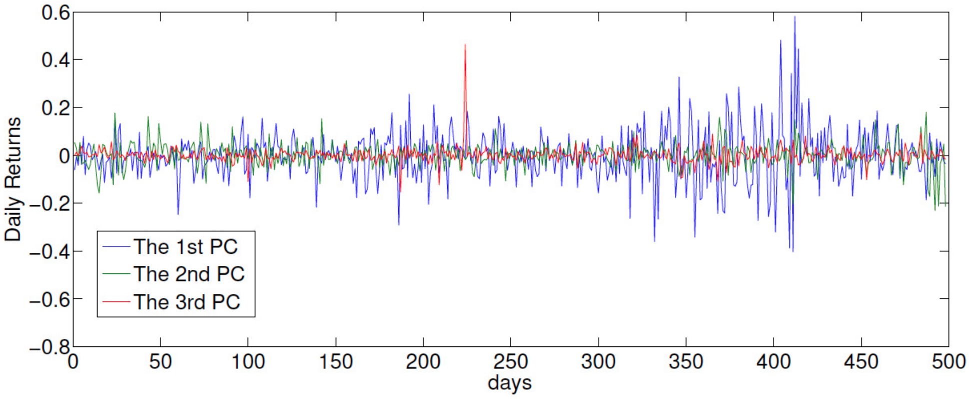

Figure 3. The first three principle components of daily log-returns

We skip description of Rissanen's 'Context' algorithm implemented by M. Mächler and P. Bühlmann in 'Context' software [4] aimed at estimation of all **contexts** (see (2)) and corresponding transition probabilities. We note only that its restriction to a maximal alphabet size ≤ 27 forces compression of our three-variate data into triples with three-valued components, which can make them less informative than the corresponding two-variate and univariate counterparts. The author of alternative PST algorithm sent us its extension to larger alphabet size, which works much slower. The proof of consistency of 'Context' under exponential mixing of stationary data is, e.g., in [16].

Classification of finite VLMC's describing protein families was considered in [2] via predicted likelihood and in [1, 5] using rather complicated statistical tests on stochastic suffix trees.

Simultaneously, the elementary nonparametric CCC-test based on Conditional Complexity of Compression was studied with mathematical theory developed in [12] as justification of its around 100 successful applications to authorship attribution of Literary Texts (LT) outlined in [15]. CCC can use a wide range of Universal Compressors (UC) [8].

We use an alternative VLMC-based Likelihood Ratio (VLMClr) nonparametric homogeneity test methodologically similar to the CCC-test. Its mathematical theory is more straightforward than that of the CCC-test under exponential mixing assumption on stationary data.

The major joint feature of the CCC-test and VLMClr is cutting the query string into slices. For CCC, slice size is chosen to avoid the UC self-adaptation to the query, which would prevent discrimination unless the slice size is sufficiently small. The main advantage of the likelihood approach based on the VLMC training is that you can choose larger slices (**no adaptation takes places**). The only slicing aim is to include a majority of its contexts and to estimate the variance of the likelihood increments.

We estimate the VLMC stochastic tree of the large stationary ergodic 'training' string T using the 'Context' software. This is then concatenated first with query slices Q_k and second, with strings S_k simulated from the training distribution of the same size as $Q_k, k = 1, \ldots, K$ (for constructing simulated strings, see algorithm in [4]).

We then find log-likelihoods $L_Q(k)$ of Q_k, $L_S(k)$ of S_k using the derived probability model of the training string and the average $\bar{D}$ of their difference D. Thus the VLMC-approach difference from the CCC-methodology is in **log-likelihood** evaluations instead of the **encoded increment lengths** evaluation. The latter only approximate log-likelihoods under certain conditions including sufficient smallness of the slice size.

Next, due to asymptotic normality of log-likelihood increments, we can compute the usual empirical variance V of $\bar{D}$ and the t-statistic t as the ratio $\bar{D}/\sqrt{V}$ with $K - 1$ degrees of freedom (df). We find K^* from the condition that $t(K^*)$ is maximal. Then, the p-value of homogeneity is evaluated for the t-distribution with $K^* - 1$ df.

2. NASDAQ DATA

We use historical Nasdaq data on multivariate daily returns for almost 498 days from April 4th 2011 to Marth 27th 2013 collected from *finance.yahoo.com* and converted into log-returns. We reduce the dimensionality of single-day returns via MatLab version of the principal component analysis (PCA) and compress the data set to the sequence of first (either two or three) Principal Components (PC) describing a major part of the data variability, see figure 2. We fit their VLMC stochastic model and apply it for discrimination between statistical properties of different parts of the data. We also compare our model with GARCH.

2.1. Results for 3 PC

First, the range of each PC is divided into three equal intervals (bins) . Triples of PC-values are compressed to triples of integers from {1,2,3} according to their belonging to corresponding bins and their triples are labeled with 26 English letters from A to Z or the * symbol. The sequence of 498 triples is thus converted into a sequence of symbols, of which we display only few:

N N N N N N E N N N N N N K K N N E N N N E K N N K N N K N N N N N N N N E N E N N N N E N N N N N N N N N N N N N N N N E E N ....

Only 8 of all 27 symbols were observed in the whole sequence.

The homogeneity t-test between 1–150 and 301–420 (quiet and volatile regions) trained on 301–420 cut into 12 slices. The t-value is -2.530118.

Table 1. Variable Length Markov Chain Training Result:

alphabet	'bdejkntw'
number of alphabet	8
number of letters	120
maximal order of Markov chain	2
context tree size	7
number of leaves	5

Table 2. Inter-log-likelihood output

-5.109994	-8.113694	-11.494689	-5.622557
-5.109994	-5.199606	-5.020382	-8.624520
-6.135120	-5.109994	-8.022345	-4.507818
-5.109994	-5.622557	-4.374287	
mean m_2= -6.21184		variance v_2=3.92183	

Table 3. Intra-log-likelihood output

-6.224733	-6.135120	-5.622557	-5.622557
-10.284486	-10.012552	-10.144724	-10.347058
-7.736400	-6.498026	-9.982415	-9.603383
mean m_1= -8.1845		variance v_1=4.15721	

2.2. Prediction accuracy

We estimate the Squared Bias to show the accuracy of our prediction by predicting 10 consecutive letters from the 131st to 140th and for each letter, the prediction is based on training preceding 130 letters. The variance of the log-likelihood of the simulated letters is 0.3947234. The Squared Bias is 0.04815128 which is 8.1 times less. This illustrates the adequacy of our prediction.

Also, we predict each of 20 letters of the quiet zone by training VLMC on 120 preceding letters. The coincidence rate of predicted and actual letter is about 75 per cent.

2.3. Results for 2 PC

First, the range of each PC is divided into five equal bins. PC-values are compressed to only 5 integers 1–5 according to their belonging to corresponding bins and their pairs are labeled with 25 English letters from A to Y similarly to 3 PC case. Only 8 of all 25 symbols were observed.

The homogeneity t-test between 1–150 and 301–420 (quiet and volatile regions) trained on 301–420 gives the t-value -2.842992.

The homogeneity t-test between 1–150 and 240–300 (two relatively quiet regions) trained on 1–150 gives the t-value -1.58671.

Table 4. Variable Length Markov Chain Training Result:

alphabet	’bcghlmqr’
number of alphabet	8
number of letters	120
maximal order of Markov chain	2
context tree size	12
number of leaves	8

Table 5. Inter-log-likelihood output

-6.223766	-11.781595	-12.728522	-9.787224
-11.865476	-8.975104	-8.161708	-12.361413
-11.060277	-11.919910	-11.287646	-15.384456
-11.030601	-12.187060	-11.395472	
mean $m_2 = -11.07668$		variance $v_2 = 4.59106$	

Table 6. Intra-log-likelihood output

-13.16789	-11.48149	-13.20084	-14.30159
-13.53945	-11.42373	-10.61972	-13.97436
-13.11824	-12.94475	-12.69096	-15.52045
mean m_1= -12.99862		variance v_1=1.81131	

Table 7. Variable Length Markov Chain Training Result:

alphabet	’bcghlm’
number of alphabet	6
number of letters	150
maximal order of Markov chain	3
context tree size	6
number of leaves	2

Table 8. Inter-log-likelihood output

-9.638182	-9.840707	-9.946484
-11.005356	-12.440440	-12.426464
	mean $m_2 = -10.88294$	variance $v_2 = 1.66718$

Table 9. Inter-log-likelihood output

-13.52255	-12.31361	-13.13173	-12.70518
-11.70747	-13.76779	-10.70303	-12.11108
-11.24829	-10.87368	-10.87293	-10.91834
-11.64226	-11.09347	-10.74125	
mean $m_1 = -11.82351$		variance $v_1 = 1.1029$	

2.4. Prediction accuracy

We estimate the Squared Bias to show the accuracy of our prediction by predicting 10 consecutive letters from the 141st to 150th (quiet zone) and for each letter, the prediction is based on training the preceding 140 letters. The variance of the log-likelihood of the simulated letters is 0.06953946. The Squared Bias is 0.04438316.

Also, we predict each of 10 letters of the quiet zone by training VLMC on 140 preceding letters. The coincidence rate of predicted and actual letter was about 55 per cent. It is less than for the 3 PCs case partly because we must predict triples of bins rather than pairs of bins.

2.5. Follow up analysis

One of the major **additional advantages of VLMClr over CCC** is its more straightforward use for the follow up estimation of contexts contributing the most to the discrimination between regions that were previously shown to be distinct.

For this aim, we propose cutting both the training and query strings into several slices for estimating the mean frequencies and their empirical variances by their direct count which approximates steady state probabilities and variances of contexts.

We obtain more transparent results of this follow up analysis than those obtained with LZ-78 in [9].

Taking into account the fact that the normalized frequency of occurrences of a context (with its frequency more than some threshold) of size n is Asymptotically Normal (AN) with variance σ_i^2, σ_i can be estimated via these frequencies in slices; $i = 0$ for training string and one for query. The normalized difference between frequencies for 0 and 1 cases is then AN with variance as sum of the above variances.

For every VLMC context of the whole Training, one can evaluate its multiplicities in $k(Tr)$ Training and $k(Qu)$ Query slices of the SAME LENGTH, their corresponding empirical means $m(Tr)$, $m(Qu)$, and empirical variances $V(Tr)$, $V(Qu)$. The usual t-statistics are

$$T(k) = (m(Tr) - m(Qu))/\sqrt{(V(Tr)/k(Tr) + V(Qu)/k(Qu))} \tag{3}$$

Choose $k(Tr)$ and $k(Qu)$ from the condition that $t(k^*(Tr), k^*(Qu))$ is maximal and slice sizes are equal. The slice size must be several (say, not less than 5) sizes of the context considered We then order t^* for different contexts starting from their maximal absolute values.

We exclude the contexts with frequencies less than a threshold in our first applications, in spite of the possibility of losing some unstable but potentially significant information contained in rare contexts.

Thus, one can find p-values for their equality and order these p-values starting with the minimal ones.

In the 2-PC case, the quiet region has the pattern L (indicating that the first PC-value is located in the second bin and the second PC-value is located in the third bin) and H (indicating that the first PC-value is located in the third bin and the second PC-value is located in the second bin) while the volatile region have the pattern B (indicating that the first PC-value is located in the first bin and the second PC-value is located in the second bin) and Q (indicating that the first PC-value is located in the fourth bin while the second PC-value is located in the second bin).

In the 3-PC case, the quiet region has the pattern N (indicating that all three PC-values are located in the second bin) while the volatile region has the pattern E (indicating that the first PC-value is located in the first bin while the second and third PC-values are located in the second bin).

2.6. Comparison with GARCH

In this subsection, we will make a comparison between our VLMC method and the GARCH model ([19], [20]) applied to two different sets of financial data.

The first data set we use is the daily log-return data of APPLE Inc. starting from Jan. 2nd, 2009 (Figure 4). By observation, we pick the volatile region (the first 450 days returns)and the quiet region (the 500th to 600th days returns) to make a comparison. We first fit the data with the GARCH(1,1) modeled using the MATLAB(R2011a) GARCH toolbox.

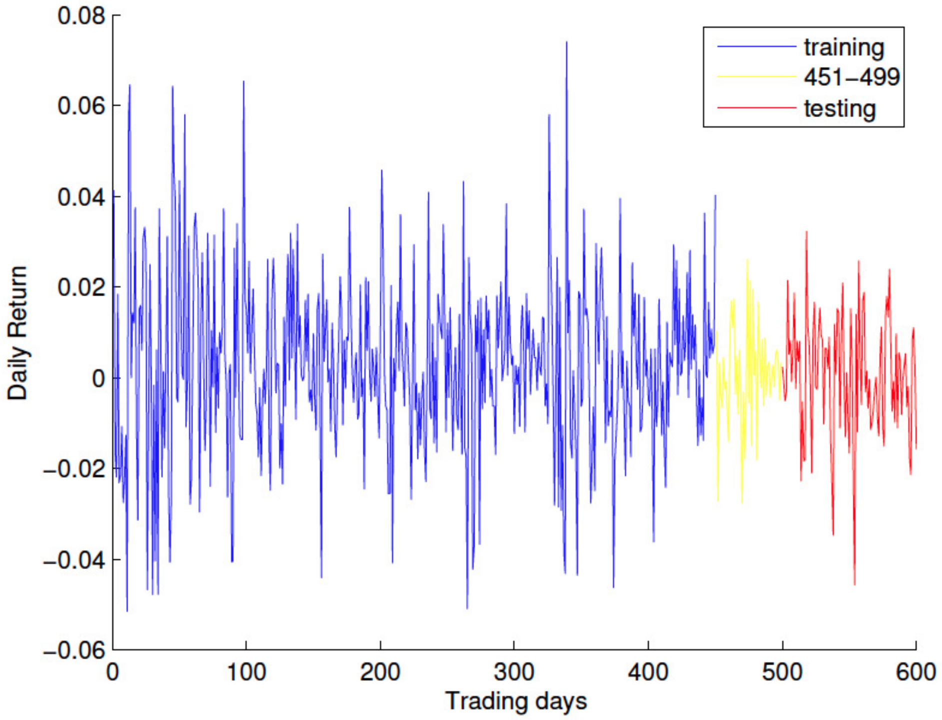

Figure 4. Apple Daily Returns

$$
\begin{aligned}
y_t &= C + \epsilon_t && (4)\\
\epsilon_t &= \sigma_t z_t && (5)\\
\sigma_t^2 &= \kappa + G_1 \sigma_{t-1}^2 + A_1 \epsilon_{t-1}^2 && (6)
\end{aligned}
$$

Let $\hat{\alpha}_1$ and $\hat{\beta}_1$ be the estimator for GARCH(1) and ARCH(1) in the first model. Similar notation can be defined for $\hat{\alpha}_2$ and $\hat{\beta}_2$. From the results, we have $z_1 = \frac{\hat{\alpha}_1 - \hat{\alpha}_2}{\sqrt{\sigma_{\alpha_1}^2 + \sigma_{\alpha_2}^2}} \doteq 2.1554$, and $z_2 = \frac{\hat{\beta}_1 - \hat{\beta}_2}{\sqrt{\sigma_{\beta_1}^2 + \sigma_{\beta_2}^2}} \doteq -1.6971$. The p-values obtained are $p_1 = 0.0311$ and $p_2 = 0.0897$.

We apply the same data on VLMC. The homogeneity t-test between 1–450 and 500–600 (quiet and volatile regions) trained on 1–450 shows that the t-value is -16.02058. Thus, the p-value $p < 0.00001$. This p-value by VLMC is much smaller than the Z-score by GARCH.

We also use the **first** principal component of Nasdaq daily log-return data for comparison with GARCH. Again, let $\hat{\alpha}_1$ and $\hat{\beta}_1$ be the estimator for GARCH(1) and ARCH(1) in the first model. Similar notation can be defined for $\hat{\alpha}_2$ and $\hat{\beta}_2$. From the results, we have $z_1 = \frac{\hat{\alpha}_1 - \hat{\alpha}_2}{\sqrt{\sigma_{\alpha_1}^2 + \sigma_{\alpha_2}^2}} \doteq -1.1798$, and $z_2 = \frac{\hat{\beta}_1 - \hat{\beta}_2}{\sqrt{\sigma_{\beta_1}^2 + \sigma_{\beta_2}^2}} \doteq 2.1554$. And thus, the p-values are $p_1 = 0.2381$ and $p_2 = 0.0311$.

We divide the range of the first PC of Nasdaq daily log-returns into 27 bins. Each bin is labeled with 26 English letters from A to Z and symbol *. The sequence of the first PC of daily log-returns is converted into a sequence of symbols. The homogeneity t-test between 1–150 and 301–420 (quiet and volatile regions) trained on 301–420 shows that the t-value is -7.048379. Thus, the p-value is $p < 0.000001$. This p-value by VLMC is much smaller than the p-value by GARCH.

3. MADISON VS HAMILTON DISCRIMINATION OF STYLES

The Federalist Papers written by Alexander Hamilton, John Jay and James Madison appeared in newspapers in October 1787–August 1788. Their intent was to persuade the citizens of the State of New York to ratify the U.S. Constitution. Seventy seven essays first appeared in several different newspapers all based in New York and then eight additional articles written by Hamilton on the same subject were published in booklet form. Ever since that time, the consensus has been that John Jay was the sole author of five (No. 2–5, No. 64) out of a total 85 papers, that Hamilton was the sole author of 51 papers (Hf), that Madison was the sole author of 14 papers (Mf, No. 10,14,37–48) and that Madison and Hamilton collaborated on another three (No. 18–20). The authorship of the remaining 12 papers (Df, No. 49–58, 62,63) has been in dispute; these papers are usually referred to as the disputed papers. It has been generally agreed that the Df-papers were written by either Madison or Hamilton, without consensus on particulars. On the other hand, [13, 7, 17] and other stylometry attributors have given all Dfs to Madison. Our goal has been to answer the following 2 questions:

1. Does VLMC-methodology attribute all Mf to Madison and reject significantly the identification of the Hf style with that of Mf?

2. What contexts are most statistically different between Mf and Hf?

The answer we have is 'yes' obtained for the first question: Mf is attributed to Madison, Hf and Mf identity of styles is rejected.

2. Our goal is to find patterns that occur in one body significantly more often than in the other. The patterns (VLMC contexts) we will examine are those that arise during the training process using the VLMC algorithm.

To generate meaningful statistics, we divide each file into *slices* of a given size, and run the 'Context' program to find the log-likelihood of each slice. We then apply the t-test to find out whether the styles of articles written by the two authors are significantly different. For each pattern, we calculate the mean number of occurrence for each file. Finally, we perform the t-value for each pattern (VLMC context).

Here is a detailed description of the algorithm:

1. For each of the two files Madison (M), Hamilton (H):
   (a) Train the VLMC model on Madison's text.
   (b) Cut both the M and H texts into slices of equal given size.
   (c) For each slice:
      i. Using Madison VLMC probability context tree to calculate the log-likelihood of each slice.
      ii. Calculate the mean value, variance of slices of M and H respectively.
      iii. Calculate the t-value as $\frac{|L_1-L_2|}{\sqrt{v_1/n_1+v_2/n_2}}$. where L_1 and L_2 are mean values of log-likelihood of slices of M and H, v_1 and v_2 are variances over slices of M and H, n_1 and n_2 are the numbers of slices of M and H respectively.
   (d) For each pattern (VLMC context):
      i. Calculate the empirical mean occurrence.
      ii. Calculate the empirical variance of occurrence.
      iii. Calculate the t-value as $\frac{|m_1-m_2|}{\sqrt{v_1/n_1+v_2/n_2}}$, where m_1 and m_2 are mean values of number of occurrence of one pattern in each slice of M and H respectively, v_1 and v_2 are variances of occurrence of M and H. n_1 and n_2 are the numbers of slices of M and H respectively,

Here is a detailed description of the data input and output:

We first combine all 14 Madison's article into one file and use it as the training data. The maximal memory n is set to be 15. (Thus at most 15 *letters* predict the next letter)

Table 10. Data from Madison

fileproc10.txt	fileproc14.txt	fileproc37.txt	fileproc38.txt
fileproc39.txt	fileproc40.txt	fileproc41.txt	fileproc42.txt
fileproc43.txt	fileproc44.txt	fileproc45.txt	fileproc46.txt
fileproc47.txt	fileproc48.txt		

Table 11. Data from Hamilton

fileproc1.txt	fileproc6.txt	fileproc7.txt	fileproc8.txt
fileproc9.txt	fileproc11.txt	fileproc12.txt	fileproc13.txt
fileproc15.txt	fileproc16.txt	fileproc17.txt	fileproc31.txt
fileproc32.txt	fileproc34.txt		

Table 12. Variable Length Markov Chain Training Results:

alphabet	’*abcdefghijklmno pqrstuvwxyz’
number of alphabet	27
number of letters	228744
maximal order of Markov chain	13
context tree size	3365
number of leaves	2353

Second, the total number of letters from Madison’s article is 228744. We cut the whole Madison data into 9 slices and each slice contains 25416 letters.

Third, we perform an intra VLMC t-test. We use each slice of Madison’s data as a query string and use the remaining 8 slices as a training string. We want to compute the log-likelihood of each query string.

Table 13. Inter-log-likelihood output

-27863.56	-28047.04	-27236.75	-26559.74
-24995.70	-27173.49	-26209.20	-25182.81
-25622.52			
	mean value $m_2 = -26543.42$		variance $v_2 = 1261004$

Fourth, we import the data from Hamilton. The total number of letters of Hamilton’s article is 152496. We cut the letters into 6 slices so that each slice contain 25416 letters (25416x6=152496)

Last, we perform an inter-VLMC t-test. We use the total training result of Madison to predict the log-likelihood of each slice of Hamilton.

Table 14. Intra-log-likelihood output

-28552.20	-28462.64	-28511.57	-28234.03
-27227.31	-26510.97		
	mean value $m_1 = -27916.45$		variance $v_1 = 721562.6$

We use the following formula for the t-test:

$$t = \frac{l_1 - l_2}{\sqrt{var_1/n_1 + var_2/n_2}}$$

plug in the numbers and we get the t-value 2.690809.

Finally, we have done a comparison "between Mf and itself (we are supposed to get a t-value around 0).

The inter VLMC of Madison vs Madison log-likelihood result:

Table 15. Inter-log-likelihood output

-27725.26	-27341.10	-26948.34	-26352.83
-23933.95	-26703.40	-25770.18	-24609.77
-25525.76			
mean $m_2 = 26101.18$		variance $v_2 = 1585075$	

The result of Inter- and Intra- VLMC t-test of Madison: the t-value is -0.7864356.

Cutting Madison's text into 14 slices and performing a t-test between Madison's and Hamilton's data by the method described above, we get the t-score 3.099237

Cutting the Madison's text into 20 slices and performing a t-test between Madison's and Hamilton's data by the method shown above, we get the t-score 4.016742

Frequencies of VLMC contexts

We have calculated frequencies of each VLMC context over slices and found patterns with the most significantly different frequencies between Madison's and Hamilton's articles. This mining if interesting for linguists can be of help to convince them in the usefulness of the statistical LT processing.

The majority of differentially expressed patterns found in [9] using LZ-78 are also found further on our larger list.

1 The following are patterns that Madison used significantly more frequently than Hamilton (we cut Madison's data into 14 slices, **symbol * denotes space**):

For p-value < 0.005

nt* , rs* , fore* , han* , de* , ed*b , by , by* , by*t , f* , nd*be , orm , *on*th , *th , *on*t , ese* , i* , over* , ay*be , und* , d* , he*n , g*the* , if* , der* , he* , ple* , ay*b , he*for , he*or , both , c* , re* , *ele , bot , ix , lt* , sume , ho* , ke* , *el , y*the* , latt , latte , oin* , *bo , *ne , by*o , veral* , *han , seco , *lat , d*th , in* , tas , ore* , eside , mer* , is* , ewe , rs*a , *se , tes* , ns* , at*th , lf* , d*on , ver*

For $0.005 \leq p$-value < 0.01

ca* , ose* , lst , *de , vera , *we , e* , nve , *cri , ere* , dur , sever , he*se , pe* , at*t , tuti , e*g , nsti , titu , w*f , nta , he*f , *tit , nin , sing , tit , ution* , rif

2 The following are patterns that Hamilton used significantly more frequently than Madison:

For p-value < 0.005

upo , pon , *k , *up , duc , ne , uis , ance , *le , om , om* , nati , wou , ne* , *at , *at* , ct , *wo , nation , id , way , *nat , ould* , e*ar , ontr , ity* , *ther , tim , qu , there , there* , *to , *to* , ut , ract , ign , e*to* , ndu , ont , ies , lit , eso , *if , erac , eng , it , it* , thin , wea , oul , dic , es*of* , *this* , ld , ld* , ilit , kind , *us , *us* , va

For $0.005 \leq p$-value < 0.01

he*i , ten , arc , rri , y*t , lig , ig , s , s* , ation , sit , *lan , comm , ld*b , time , uisi , ces , iv

We also did this analysis by cutting into 9 and 20 slices. The results are available by request.

4. HELIUM EMISSIONS AND SEISMIC EVENTS

Complex processes in the Earth's crust preceding strong earthquakes imply changes of concentrations of certain chemicals dissolved in ground water samples from deep wells. Of particular interest in the region of Armenia studied in [10] are the Helium emissions into the water samples from deep wells (unlike studies in [11] of near–surface emissions) that appear attractive as a potential earthquake predictor.

We consider an approximately 10-year–long set of Helium emissions data during the nineteen eighties and nineties from three deep Armenian wells in Kadaran, Ararat and Surenavan and the earthquake dates in their vicinity shown in our figure 4 were sent to us by Dr. E.A. Haroutunian (Inst. for Informatics and Automat. Problems, Armenian Acad. Sci.) for our robust analysis. In [10] they showed separately for each well that the Wilcoxon statistical test distinguishes between quiet region of the plot and that preceding strong earthquakes. The Wilcoxon test was derived under assumption of samples independence, which does not hold in this application. Our problem was to check if VLMClr can distinguish between the regions given above. Instead of separate studies of data from the three wells we used PCA–compressed data. After the observations started, the earthquake days were 529, 925, 1437, 1797, 1997, 2470, 2629 and 2854. The singular value plot (figure 6) suggests using either one or 2 PCs.

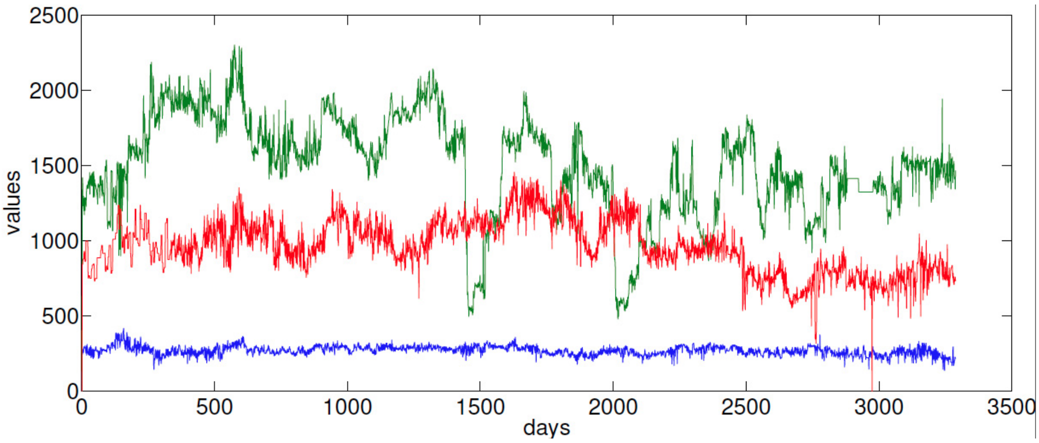

Figure 5. Helium emissions data from three deep Armenian wells

In the one–PC case, the range is divided into 27 equal bins. PC-values are labeled with 26 English letters from A to Z and symbol * according to their belonging to bins. The 'Context' gave us the following parameters of the stochastic context tree:

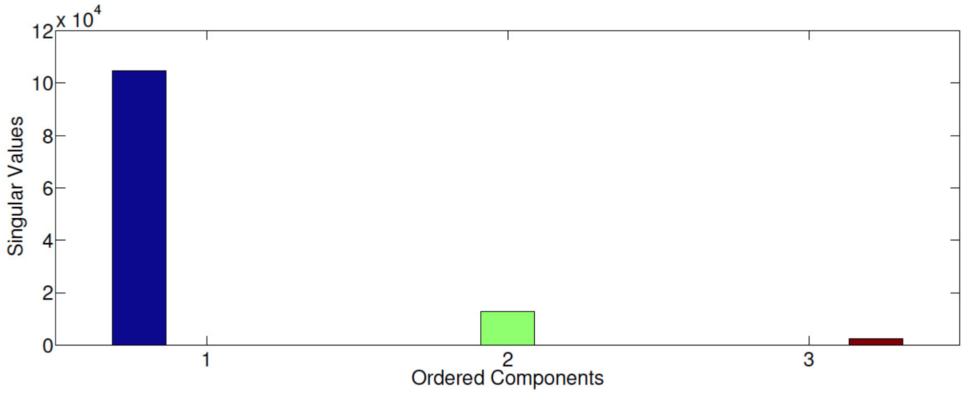

Figure 6. Singular Values

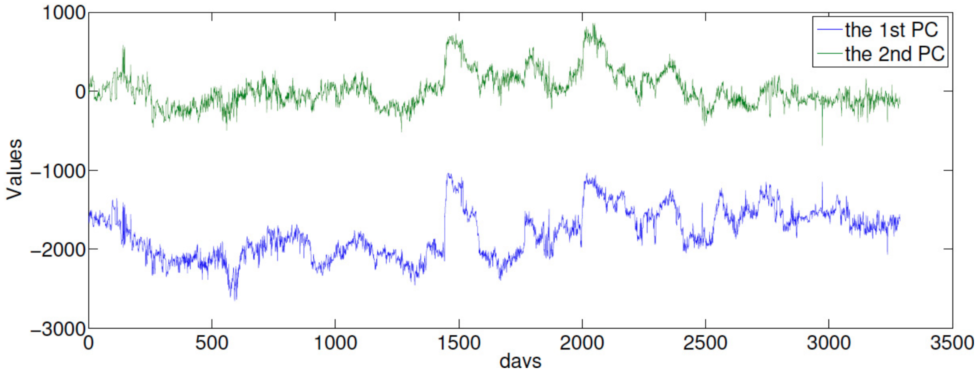

Figure 7. Top two principle components

Table 16. Variable Length Markov Chain Training Result:

alphabet	'abfghlmnqrstw'
number of alphabet	17
number of letters	400
maximal order of Markov chain	3
context tree size	28
number of leaves	21

The quiet region 1–400 has the letters "hijklmnopqr while the volatile region before and after earthquake 429–568 has the letters "opqrstuvwxy". It is impossible to train either region to predict the other one. It is also not necessary to do that because one can easily distinguish between different regions by observing that the quiet region has the first PC value up above a level "n"(corresponding to a value -1815.101), and the volatile region has the first PC value down below a level "s"(corresponding to a number -2174.363).

In the 2–PCs case, the range of each PC is divided into 5 equal bins. PC-values are compressed to 5 integers according to their belonging to bins and their pairs are labeled with 25 English letters from A to Y. We just list the first few letters as an example of labeled letters:

r r r r r r r r r r r r r r r r r r r m m m m q q q q q q q q r q q q q q r q q r r m m r m r r r q q q q q q q q r r r r r r r r g g g...

Training the quiet region between 1–400 will provide the following training result:

The homogeneity t-value between 1–400 (quiet region) and 429–528 (before earthquake) is 4.4 which means that these two region are quite different. By calculating t-value for each context, we

Table 17. Variable Length Markov Chain Training Result:

alphabet	'abfghlmnqrstw'
number of alphabet	13
number of letters	400
maximal order of Markov chain	3
context tree size	22
number of leaves	15

get the context that distinguishes the most between these two regions. "l"(the first PC value in the third bin and the second PC value in the second bin) is the typical pattern of volatile regions before earthquake and "b"(the first PC value is in the first quartile and the second PC value is in the second quartile) is the typical pattern of quiet region.

The homogeneity t-test between 1–400 (quiet region and 529–568 (*after* earthquake) is 0.8, which means that we find not much difference between the quiet region and the region after earthquake. In addition, we find an interesting letter "c"(it means that the first PC is located in the first bin and the second PC is in the third bin) which can be an indicator for quiet times to follow because, when each "c"appears, there were at least 100 quiet days beyond it in the future.

5. DISCUSSION AND CONCLUSIONS

Our results show that the VLMC modeling can be useful in various prediction and inference problems of certain time series.

Our next goal will be to use versions of the 'Context' program including the PST for enabling processing larger alphabet faster using parallel computing and running on line versions of stochastic context tree construction for the fast on line detection of abrupt changes in data statistical profiles and for processing musical scores.

ACKNOWLEDGEMENT

The authors are deeply indebted to Dr. E.A. Haroutunian (Inst. for Informatics and Automat. Problems, Armenian Acad. Sci.) for sending us data and papers on Radon emissions and seismology, to Dmitry Malioutov (IBM Watson Research) and to Libin Cheng (Morgan Stanley, Inc.) for his stimulating discussion of our first case study. We acknowledge kind useful style remarks of Prof. E. Gover and Robert Chase (Northeastern University).

REFERENCES

1. Balding D., Ferrari P. A. , Fraiman R., Sued M. *Limit theorems for sequences of random trees.* arXiv.org > stat > arXiv:math/0406280, 2007.

2. Bejerano G. *Automata learning and stochastic modeling for biosequence analysis.* PhD dissertation, Jerusalem: Hebrew University, 2003.

3. Belloni A. and Oliveira R. I. *Approximate group context tree: applications to dynamic programming and dynamic choice models.* arXiv.org > stat > arXiv:1107.0312, 2011.

4. Mächler M. and Bühlmann P. Variable Length Markov Chains: methodology, computing, and software. *Journal of Computational and Graphical Statistics*, 2004, vol. 13, no. 2, pp. 435 – 455.

5. Busch J.R., Ferrari P.A., Flesia A.G., Fraiman R., Grynberg S.P., and Leonardi F. Testing statistical hypothesis on random trees and applications to the protein classification problem, *The Annals of Applied Statistics*, 2009, vol. 3, no. 2, pp. 542–563.

6. Malyutov M.B. Authorship attribution of literary texts: a review. *Review of Applied and Industrial Mathematics*, 2005, vol. 12, no. 1, pp. 41–77 (in Russian).

7. Malyutov M.B., Wickramasinghe C.I., and Li S. *Conditional Complexity of Compression for Authorship Attribution*, SFB 649, Discussion paper no. 57, Humboldt University. Berlin, 2007.

8. Cover T.M. and Thomas J.A. *Elements of information theory.* Hoboken: Wiley, 2006, second edition.

9. Malyutov M.B. and Cunningham G. LZ-78 generated patterns in texts inhomogeneity, *International Conference on Computational Technologies in Electrical and Electronics Engineering, IEEE Region 8, SIBIRCON 2010, Proceedings.* 2010, vol. 1, pp. 15–22, available via IEEXplore.

10. Haroutunian E.A., Safarian I.A., Petrossian P.A., Nersesian H.V. Earthquake precursor identification on the base of statistical analysis of hydrogeochemical time series. *Mathematical problems of Computer Science*, 1997, vol. 18, pp. 33–39.

11. Reimer G.M. Use of soil-gas helium concentrations for earthquake prediction: Limitations Imposed by Diurnal Variation. *Journal of Geophysical Research*, 1980, vol. 85, no. B6, pp. 3107–3114.

12. Malyutov M.B. Compression based homogeneity testing. *Doklady of Russian Acad. Sci.*, 2012, vol. 443, no. 4, pp. 427–430.

13. Mosteller F. and Wallace D. *Inference and disputed authorship: The Federalist papers.* Reading: Addison-Wesley, 1964.

14. Rissanen J. A universal data compression system. *IEEE Trans. Inform. Theory.* 1983, vol. 29, no. 5, pp. 656–664.

15. Ryabko B., Astola J., Malyutov M.B. *Compression-based methods of prediction and statistical snalysis of time series: theory and applications.* TICSP series no. 56. Tampere Technical University. Tampere, 2010.

16. Galves A. and Loecherbach E. Stochastic chains with memory of variable length, In: *Festschrift in Honor of Jorma Rissanen on the Occasion of his 75th Birthday*, TICSP series No. 38. Tampere: Tampere Tech. Uni., 2008, pp. 117–134.

17. Wickramasinghe C. I. *The relative conditional complexity of compression for authorship attribution of texts.* PhD dissertation, Boston, MA: Mathematics Department, Northeastern University, 2005.

18. Ziv J. *A Note on the Compaction of long Training Sequences for Universal Classification – a Non–Probabilistic Approach.* arxiv.org/abs/1102.5482, 2012.

19. *GARCH toolbox User's guide.* The MathWorks, Inc, 2002.

20. Hull J. C. L. *Options, futures, and other derivatives.* Prentice Hall, 2011, 8th edition.