Decision-tree Induction from Time-series Data Based on a Standard-example Split Test

Yuu Yamada YUU@SLAB.DNJ.YNU.AC.JP
Einoshin Suzuki SUZUKI@YNU.AC.JP
Electrical and Computer Engineering, Yokohama National University, 79-5, Tokiwadai, Hodogaya, Yokohama, 240-8501, Japan

Hideto Yokoi YOKOIH@TELEMED.HO.CHIBA-U.AC.JP
Katsuhiko Takabayashi TAKABA@HO.CHIBA-U.AC.JP
Division of Medical Informatics, Chiba University Hospital, 1-8-1 Inohana, Chuo-ku, Chiba, 260-8677 Japan

Abstract

This paper proposes a novel decision tree for a data set with time-series attributes. Our time-series tree has a value (i.e. a time sequence) of a time-series attribute in its internal node, and splits examples based on dissimilarity between a pair of time sequences. Our method selects, for a split test, a time sequence which exists in data by exhaustive search based on class and shape information. Experimental results confirm that our induction method constructs comprehensive and accurate decision trees. Moreover, a medical application shows that our time-series tree is promising for knowledge discovery.

1. Introduction

A decision tree (Breiman, Friedman, Olshen, & Stone, 1984; Murthy, 1998; Quinlan, 1993) represents a tree-structured classifier which performs a split test in its internal node and predicts a class of an example in its leaf node. Various algorithms for decision-tree induction have been successful in numerous application domains. A basic split test assumes a nominal attribute, and allocates examples according to their attribute values to their corresponding child nodes. In order to apply decision-tree induction methods to various problems, the basic split test has been extended for numerical attributes, tree-structured attributes (Almuallim, Akiba, & Kaneda, 1996), and set-valued attributes (Takechi & Suzuki, 2002). A tree-structured attribute takes a value in a hierarchy, and a set-valued attribute takes a set as its value.

Time-series data consist of a set of time sequences each of which represents a list of values sorted in chronological order, and are abundant in various domains (Keogh, 2001). Conventional classification methods for such data can be classified into a transformation approach and a direct approach. The former maps a time sequence to another representation. The latter, on the other hand, typically relies on a dissimilarity measure between a pair of time sequences. They are further divided into those which handle time sequences that exist in data (Rodríguez, Alonso, & Bostrvm, 2000) and those which rely on abstracted patterns (Drücker et al., 2002; Geurts, 2001; Kadous, 1999).

Comprehensiveness of a classifier is highly important in various domains including medicine. The direct approach, which explicitly handles real time sequences, has an advantage over other approaches. In our chronic hepatitis domain, we have found that physicians tend to prefer real time sequences instead of abstracted time sequences which can be meaningless. However, conventional methods such as (Rodríguez, Alonso, & Bostrvm, 2000) rely on sampling and problems related with extensive computation in such methods still remained unknown.

In this paper, we propose, for decision-tree induction, a split test which finds the "best" time sequence that exists in data with exhaustive search. A time-series tree represents a novel decision tree which employs this sort of split tests and dynamic time warping (DTW)

Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

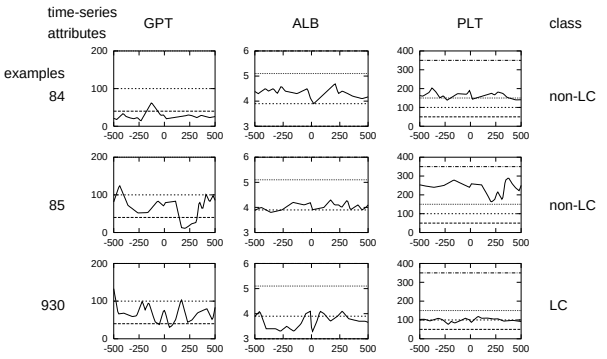

Figure 1. Data set which consists of time-series attributes

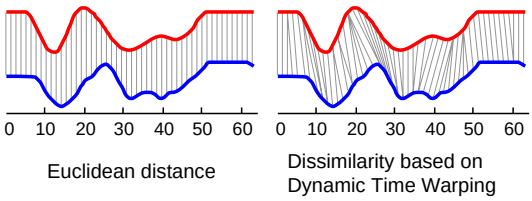

Figure 2. Correspondence of a pair of time sequences in the Euclidean distance and the DTW-based measure (Artwork courtesy of Eamonn Keogh)

(Sakoe & Chiba, 1978) in our dissimilarity measure.

2. Classification from Time-series Data

2.1. Definition of the Problem

A time sequence $\boldsymbol{A}$ represents a list of values $\alpha_1, \alpha_2, \cdots, \alpha_I$ sorted in chronological order. For simplicity, this paper assumes that the values are obtained or sampled with an equivalent interval (=1)[1].

A data set D consists of n examples e_1, e_2, $\cdots$, e_n, and each example e_i is described by m attributes a_1, a_2, $\cdots$, a_m and a class attribute c. We assume that an attribute a_j represents a time-series attribute which takes a time sequence as its value[2]. The class attribute c represents a nominal attribute and its value is called a class. We show an example of a data set which consists of time-series attributes in Figure 1.

In classification from time-series data, the objective represents induction of a classifier, which predicts the class of an example e, given a training data set D. This paper mainly assumes a decision tree (Breiman et al., 1984; Murthy, 1998; Quinlan, 1993) as a classifier.

2.2. Dissimilarity Measure Based on Dynamic Time Warping

In this section, we define two dissimilarity measures for a pair of time sequences. The first measure represents a Manhattan distance, which is calculated by taking all the vertical differences between a pair of data points of the time sequences, and then summing their absolute values together[3]. It should be noted that Manhattan distance as well as Euclidean distance cannot be applied to a pair of time sequences with different numbers of values, and the results can be counter-intuitive (Keogh & Pazzani, 2000). The main reason lies in the fact that these measures assume a fixed vertical correspondence of values, while a human being can flexibly recognize the shape of a time sequence.

The second measure is based on Dynamic Time Warping (DTW) (Sakoe & Chiba, 1978). DTW can afford non-linear distortion along the time axis since it can assign multiple values of a time sequence to a single value of the other time sequence. This measure, therefore, can not only be applied to a pair of time sequences with different numbers of values, but also fits human intuition. Figure 2 shows examples of correspondence in which the Euclidean distance and the dissimilarity measure based on DTW are employed. From the Figure, we see that the right-hand side seems more natural than the other.

Now we define the DTW-based measure $G(\boldsymbol{A}, \boldsymbol{B})$ between a pair of time sequences $\boldsymbol{A} = \alpha_1, \alpha_2, \cdots, \alpha_I$ and $\boldsymbol{B} = \beta_1, \beta_2, \cdots, \beta_J$. The correspondence between $\boldsymbol{A}$ and $\boldsymbol{B}$ is called a warping path, and can be represented as a sequence of grids $\boldsymbol{F} = f_1, f_2, \ldots, f_K$ on an $I \times J$ plane as shown in Figure 3.

Let the distance between two values α_{i_k} and β_{j_k} be $d(f_k) = |\alpha_{i_k} - \beta_{j_k}|$, then an evaluation function $\Delta(\boldsymbol{F})$ is given by $\Delta(\boldsymbol{F}) = 1/(I+J)\sum_{k=1}^{K} d(f_k)w_k$. The smaller the value of $\Delta(\boldsymbol{F})$ is, the more similar $\boldsymbol{A}$ and $\boldsymbol{B}$ are. In order to prevent excessive distortion, we assume an adjustment window ($|i_k - j_k| \leq r$), and consider minimizing $\Delta(\boldsymbol{F})$ in terms of $\boldsymbol{F}$, where w_k is a positive weight for f_k, $w_k = (i_k - i_{k-1}) + (j_k - j_{k-1})$, $i_0 = j_0 = 0$. The minimization can be resolved without checking all possible $\boldsymbol{F}$ since dynamic programming, of which complexity is $O(IJ)$, can be employed. The minimum value of $\Delta(\boldsymbol{F})$ gives the value of $G(\boldsymbol{A}, \boldsymbol{B})$.

[1] This restriction is not essential in DTW.

[2] Though it is straightforward to include other kinds of attributes in the definition, we exclude them for clarity.

[3] We selected this instead of Euclidean distance hoping for its robustness against outliers. Experiments, however, showed that there is no clear winner in accuracy.

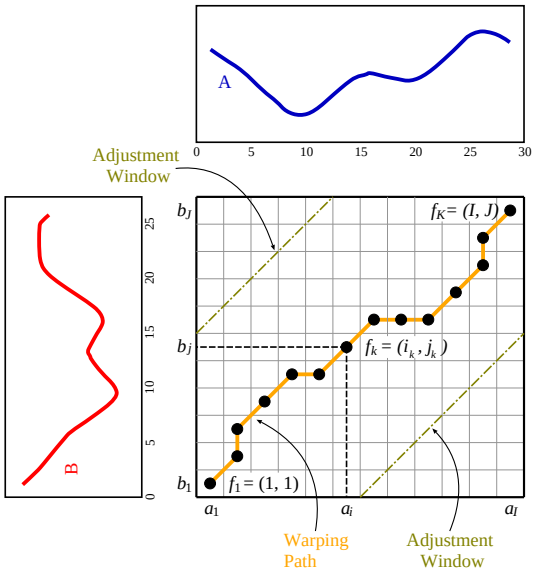

Figure 3. Example of a warping path (Artwork courtesy of Eamonn Keogh)

2.3. Problems of Conventional Methods

In (Kadous, 1999), a feature extraction approach is adopted to handle time series data with a conventional classification algorithm. As we stated in section 1, we believe that the transformation approach produces less comprehensive classifiers than the direct approach. (Drücker et al., 2002; Geurts, 2001) generate abstracted time sequences each of which is employed in a classifier. As we will see in section 4.3, real time sequences which exist in data are favored in our chronic hepatitis domain. (Rodríguez, Alonso, & Bostrvm, 2000) selects real time sequences in a decision list based on class information, but relies on a sampling method. As we will see in section 3.1, exhaustive search improves simplicity of a classifier and shapes of time sequences should be also considered in the learning process.

Naive methods for classification from time-series data include a decision-tree learner which replaces a time sequence with its average value in pre-processing and a nearest neighbor method with the DTW-based dissimilarity measure. The former neglects the structure of a time sequence, and might possibly recognize a pair of largely different time sequences as similar. The latter, as a lazy learner, outputs no classification model thus its learning results lack of comprehensibility.

3. Time-series Tree

3.1. Standard-example Split Test

In order to circumvent the aforementioned problems, our time-series tree has a time sequence which exists in data and an attribute in its internal node, and splits a set of examples according to the dissimilarity of their corresponding time sequences to the time sequence. The use of a time sequence which exists in data in its split node contributes to comprehensibility of the classifier, and each time sequence is obtained by exhaustive search. The dissimilarity measure is based on DTW due to the reasons described in section 2.2.

We call this split test a standard-example split test. A standard-example split test $\sigma(e, a, \theta)$ consists of a standard example e, an attribute a, and a threshold θ. Let a value of an example e in terms of a time-series attribute a be $e(a)$, then a standard-example split test divides a set of examples $e_1, e_2, \cdots, e_n$ to a set $S_1(e, a, \theta)$ of examples each of which $e_i(a)$ satisfies $G(e(a), e_i(a)) < \theta$ and the rest $S_2(e, a, \theta)$. We also call this split test a θ-guillotine cut.

As the goodness of a split test, we have selected gain ratio (Quinlan, 1993) since it is frequently used in decision-tree induction. Since at most $n-1$ split points are inspected for an attribute in a θ-guillotine cut and we consider each example as a candidate of a standard example, it frequently happens that several split points exhibit the largest value of gain ratio. We assume that shapes of time sequences are essential in classification, thus, in such a case, we define that the best split test exhibits the largest gap between the sets of time sequences in the child nodes. The gap $gap(e, a, \theta)$ of $\sigma(e, a, \theta)$ is equivalent to $G(e'(a), e''(a))$ where e' and e'' represent the example $e_i(a)$ in $S_1(e, a, \theta)$ with the largest $G(e(a), e_i(a))$ and the example $e_j(a)$ in $S_2(e, a, \theta)$ with the smallest $G(e(a), e_j(a))$ respectively. When several split tests exhibit the largest value of gain ratio, the split test with the largest $gap(e, a, \theta)$ among them is selected.

Below we show the procedure *standardExSplit* which obtains the best standard-example split test, where ω.gr and ω.gap represent the gain ratio and the gap of a split test ω respectively.

Procedure: *standardExSplit*
Input: Set of examples $e_1, e_2, \cdots, e_n$
Return value: Best split test ω
1 ω.gr $= 0$
2 **Foreach**(example e)
3 **Foreach**(time-series attribute a)
4 Sort examples $e_1, e_2, \cdots, e_n$ in the current node using $G(e(a), e_i(a))$ as a key to $\epsilon_1, \epsilon_2, \cdots, \epsilon_n$

```
5       Foreach(θ-guillotine cut ω′ of ϵ1, ϵ2, ···, ϵn)
6         If ω′.gr > ω.gr
7           ω = ω′
8         Else If ω′.gr == ω.gr And ω′.gap > ω.gap
9           ω = ω′
10  Return ω
```

3.2. Cluster-example Split Test

From the standard-example split test, one can easily invent a split test with two standard examples e', e'' and an attribute a. A cluster-example split test $\sigma'(e', e'', a)$ divides a set of examples $e_1, e_2, \cdots, e_n$ into a set $U_1(e', e'', a)$ of examples each of which $e_i(a)$ satisfies $d(e'(a), e_i(a)) < d(e''(a), e_i(a))$ and the rest $U_2(e', e'', a)$. The evaluation criterion for the goodness of a split test is equivalent to that of the standard-example split test without θ.

Below we show the procedure *clusterExSplit* which obtains the best cluster-example split test.

Procedure: *clusterExSplit*
Input: Set of examples $e_1, e_2, \cdots, e_n$
Return value: Best split test ω

```
1   ω.gr = 0
2   Foreach(pair of examples e′, e″)
3     Foreach(time-series attribute a)
4       ω′ = σ′(e′, e″, a)
6       If ω′.gr > ω.gr
7         ω = ω′
8       Else If ω′.gr == ω.gr And ω′.gap > ω.gap
9         ω = ω′
10  Return ω
```

4. Experimental Evaluation

4.1. Conditions of Experiments

In order to investigate the effectiveness of the proposed method, we perform experiments with real-world data sets. Chronic hepatitis data have been donated from Chiba University Hospital in Japan, and have been employed in (Berka, 2002). The Australian sign language data[4] and the EEG data belong to the benchmark data sets in the UCI KDD archive (Hettich & Bay 1999).

For the chronic hepatitis data, we have settled a classification problem of predicting whether a patient is liver cirrhosis (the degree of fibrosis is F4 or the result of the corresponding biopsy[5] is LC in the data). Since the patients underwent different numbers of medical tests, we have employed time sequences each of which has more than 9 test values during a period of before 500 days and after 500 days of a biopsy. As consequence, our data set consists of 30 LC-patients and 34 non-LC patients. Since the intervals of medical tests differ, we have employed liner interpolation between two adjacent values and transformed each time sequence to a time sequence of 101 values with a 10-day interval. One of us, who is a physician, suggested to use in classification 14 attributes (GOT, GPT, ZTT, TTT, T-BIL, I-BIL, D-BIL, T-CHO, TP, ALB, CHE, WBC, PLT, HGB) which are important in hepatitis. We considered that the change of medical test values might be as important as the values, and have generated 14 novel attributes as follows. Each value $e(a, t)$ of an attribute a at time t is transformed to $e'(a, t)$ by $e'(a, t) = e(a, t) - \{l(e, a) + s(e, a)\}/2$, where $l(e, a)$ and $s(e, a)$ represent the maximum and the minimum value of a for e respectively. As consequence, another data set with 28 attributes was used in the experiments.

The Australian sign language data represent a record of hand position and movement of 95 words which were uttered several times by five people. In the experiments, we have chosen randomly five words "Norway", "spend", "lose", "forget", and "boy" as classes, and employed 70 examples for each class. Among the 15 attributes, 9 (x, y, z, roll, thumb, fore, index, ring, little) are employed and the number of values in a time sequence is 50[6]. The EEG data set is a record of brain waves represented in a set of 255-value time sequences obtained from 64 electrodes placed on scalps. We have agglomerated three situations for each patient and omitted examples with missing values, and obtained a data set of 192 attributes for 77 alcoholic people and 43 non-alcoholic people.

In the experiments, we tested two decision-tree learners with the standard-example split test (SE-split) and the cluster-example split test (CE-spit) presented in section 3. Throughout the experiments, we employed the pessimistic pruning method (Mingers, 1989) and the adjustment window r of DTW was settled to 10 % of the total length[7].

For comparative purpose, we have chosen methods presented in section 2.3. In order to investigate on the effect of our exhaustive search and use of gaps, we tested a modified versions of SE-split which select each standard example from randomly chosen 1 or 10 examples without using gaps as Random 1 and Random

[4]The original source is http://www.cse.unsw.edu.au/~waleed/tml/data/.

[5]We will explain a biopsy in section 4.3.

[6]For each time sequence, 50 points with an equivalent interval were generated by linear interpolation between two adjacent points.

[7]A value less than 1 is round down.

10 respectively. Intuitively, the former corresponds to a decision-tree version of (Rodríguez, Alonso, & Bostrvm, 2000). In our implementation of (Geurts, 2001), the maximum number of segments was set to 3 since it outperformed the cases of 5, 7, 11. In our implementation of (Kadous, 1999), we used average, maximum, minimum, median, and mode as global features, and Increase, Decrease, and Flat with 3 clusters and 10-equal-length discretization as its "parametrised event primitives". Av-split, and 1-NN represent the split test for the average values of time sequences, and the nearest neighbor method with the DTW-based measure respectively.

It is widely known that the performance of a nearest neighbor method largely depends on its dissimilarity measure. As the result of trial and error, the following dissimilarity measure $H(e_i, e_j)$ has been chosen since it often exhibits the highest accuracy.

$$H(e_i, e_j) = \sum_{k=1}^{m} \frac{G(e_i(a_k), e_j(a_k))}{q(a_k)},$$

where $q(a_k)$ is the maximum value of the dissimilarity measure for a time-series attribute a_k, i.e. $q(a_k) \geq \forall i \forall j \ G(e(a_i), e(a_j))$.

In the standard-example split test, the value of a gap $gap(e, a, \theta)$ largely depends on its attribute a. In the experiments, we have also tested to substitute $gap(e, a, \theta)/q(a)$ for $gap(e, a, \theta)$. We omit the results in the following sections since this normalization does not necessarily improve accuracy.

4.2. Results of the Experiments

We show the results of the experiments in Tables 1 and 2 where the evaluation methods are leave-one-out and 20 × 5-fold cross validation respectively. In the Tables, the size represents the average number of nodes in a decision tree, and the time is equal to {(the time for obtaining the DTW values of time sequences for all attributes and all pairs of examples) + (the time needed for leave-one-out or 20 times 5-fold cross validation)}/(the number of learned classifiers) in order to compare eager learners and a lazy learner. A PC with a 3-GHz Pentium IV CPU with 1.5G-byte memory was used in each trial. H1 and H2 represent the data sets with 14 and 28 attributes respectively both obtained from the chronic hepatitis data described in section 4.1.

First, we briefly summarize comparison of our SE-split with other non-naive methods. From the tables, our exhaustive search and use of gaps are justified since SE-split draws with Random 1 and 10 in accuracy but outperforms them in terms of tree sizes. Our SE-split draws with (Geurts, 2001) in terms of these criteria but is much faster[8]. We have noticed that (Geurts, 2001) might be vulnerable to outliers but this should be confirmed by investigation. We attribute the poor performance of our (Kadous, 1999) to our choice of parameter values, and consider that we need to tune them to obtain satisfactory results.

In terms of accuracy, Random 1, (Kadous, 1999), and Av-split suffer in the EEG data set and the latter two in the Sign data set. This would suggest effectiveness of exhaustive search, small number of parameters, and explicit handling of time sequences. 1-NN exhibits high accuracy in the Sign data set and relatively low accuracy in the EEG data set. These results might come from the fact that all attributes are relevant in the sign data set while many attributes are irrelevant in the EEG data set.

CE-spit, although it handles the structure of a time sequence explicitly, almost always exhibits lower accuracy than SE-split. We consider that this is due to the fact that CE-split rarely produces pure child nodes[9] since it mainly divides a set of examples based on their shapes. We have observed that SE-split often produces nearly pure child nodes due to its use of the θ-guillotine cut.

Experimental results show that DTW is sometimes time-inefficient. It should be noted that time is less important than accuracy in our problem. Recent advances such as (Keogh, 2002), however, can significantly speed up DTW.

In terms of the size a decision tree, SE-split, CE-split, and (Geurts, 2001) constantly exhibit good performance. It should be noted that SE-split and CE-split show similar tendencies though the latter method often produces slightly larger trees possibly due to the pure node problem. As we have described in section 2.3, a nearest neighbor method, being a lazy learner, has deficiency in comprehensiveness of its learned results. This deficiency can be considered as crucial in application domains such as medicine where interpretation of learned results is highly important.

The difference between Tables 1 and 2 in time mainly comes from the number of examples in the training set and the test set. The execution time of 1-NN, being a lazy learner, is equivalent to the time of the test

[8] We could not include the results of the latter for two data sets since we estimate them several months with our current implementation. Reuse of intermediate results, however, would significantly shorten time.

[9] A pure child node represents a node with examples belonging to the same class.

Table 1. Results of experiments with leave-one-out

	accuracy (%)				time (s)				size			
method	H1	H2	Sign	EEG	H1	H2	Sign	EEG	H1	H2	Sign	EEG
SE-split	79.7	85.9	86.3	70.0	0.8	1.4	63.3	96.8	9.0	7.1	38.7	16.6
Random 1	71.9	68.8	85.7	54.2	0.2	0.5	1.2	50.9	15.6	12.9	69.5	33.7
Random 10	79.7	82.8	86.3	67.5	0.4	0.8	3.4	59.9	10.2	9.8	50.8	20.6
Geurts	75.0	78.1	-	-	26.2	41.1	-	-	10.2	9.5	-	-
Kadous	68.8	68.8	36.9	10.8	2.0	4.9	10.6	1167.0	9.2	10.8	39.5	40.0
CE-spit	65.6	73.4	85.4	63.3	1.3	1.3	1876.4	1300.5	9.4	7.2	42.8	23.4
Av-split	73.4	70.3	35.7	52.5	0.0	0.1	0.2	2.9	10.9	11.4	47.4	61.9
1-NN	82.8	84.4	97.8	60.8	0.2	0.4	0.1	47.5	N/A	N/A	N/A	N/A

Table 2. Results of experiments with 20 × 5-fold cross validation

	accuracy (%)				time (s)				size			
method	H1	H2	Sign	EEG	H1	H2	Sign	EEG	H1	H2	Sign	EEG
SE-split	71.1	75.6	85.9	63.8	0.5	0.8	28.3	51.1	8.3	7.5	28.3	13.4
Random 1	67.3	69.3	81.0	55.5	0.1	0.3	0.8	48.1	12.1	10.7	59.0	24.6
Random 10	73.8	72.2	84.2	60.5	0.2	0.5	2.6	64.5	9.0	8.4	41.1	16.5
Geurts	68.7	72.1	-	-	7.7	14.5	-	-	7.9	8.1	-	-
Kadous	68.1	67.7	36.2	41.1	2.1	4.7	10.4	642.9	7.9	8.7	33.5	26.8
CE-spit	69.1	69.0	86.2	58.5	0.6	1.5	966.9	530.0	7.9	7.4	36.0	19.9
Av-split	73.0	70.7	34.9	51.3	0.0	0.0	0.2	2.5	10.1	10.0	41.1	40.1
1-NN	80.9	81.5	96.6	61.8	2.2	4.4	9.3	1021.9	N/A	N/A	N/A	N/A

phase and is roughly proportional to the number of test examples. As the result, it runs faster with leave-one-out than 20 × 5-fold cross validation, which is the opposite tendency of a decision-tree learner.

The accuracy of SE-split often degrades substantially with 20 × 5-fold cross validation compared with leave-one-out. We attribute this to the fact that a "good" example, which is selected as a standard example if it is in a training set, belongs to the test set in 20 × 5-fold cross validation more frequently than in leave-one-out. In order to justify this assumption, we show the learning curve[10] of each method for EEG data, which consists of the largest number of examples in the four data sets, in Figure 4. In order to increase the number of examples, we counted a situation of a patient as an example. Hence an example is described by 64 attributes, and we have picked up 250 examples from each class randomly. We omitted several methods due to their performance in Tables 1 and 2. From the Figure, the accuracy of the decision-tree learner with the standard-example split test degrades heavily when the number of examples is small. These experiments show that our standard-example split test is appropriate for a data set with a relatively large number of examples.

[10]Each accuracy represents an average of 20 trials.

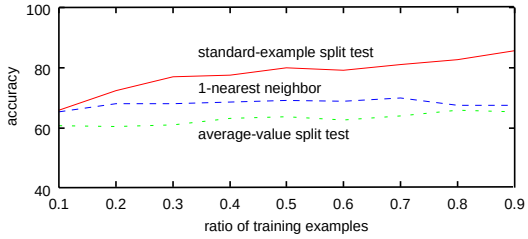

Figure 4. Learning curve of each method for the EEG data

4.3. Knowledge Discovery from the Chronic Hepatitis Data

Chronic hepatitis represents a disease in which liver cells become inflamed and harmed by virus infection. In case the inflammation lasts a long period, the disease comes to an end which is called a liver cirrhosis. During the process to a liver cirrhosis, the degree of fibrosis represents an index of the progress, and the degree of fibrosis consists of five stages ranging from F0 (no fiber) to F4 (liver cirrhosis). The degree of fibrosis can be inspected by biopsy which picks liver tissue by inserting an instrument directly into liver. A biopsy, however, cannot be frequently performed since it requires a short-term admission to a hospital and involves danger such as hemorrhage. Therefore, if a conventional medical test such as a blood test can pre-

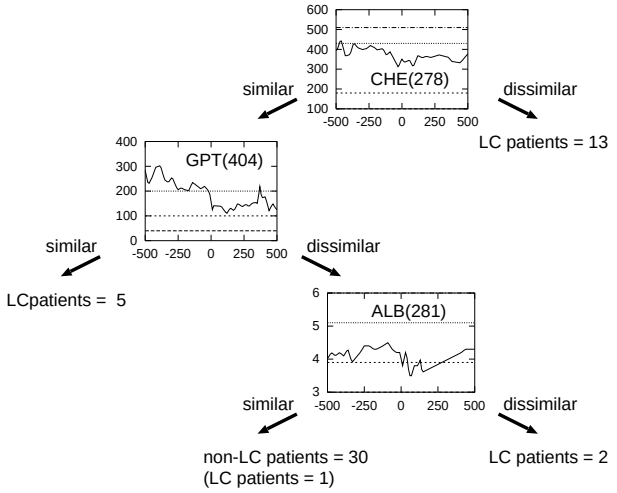

Figure 5. Time-series tree learned from H0 (chronic hepatitis data of the first biopsies)

dict the degree of fibrosis, it would be highly beneficial in medicine since it can replace a biopsy.

The classifier obtained below might realize such a story. We have prepared another data set, which we call H0, from the chronic hepatitis data by dealing with the first biopsies only. H0 consists of 51 examples (21 LC patients, and 30 non-LC patients) each of which is described with 14 attributes. Since a patient who underwent a biopsy is typically subject to various treatments such as interferon, the use of the first biopsies only enables us to analyze a more natural stage of the disease than in H1 and H2. We show the decision tree with the standard-example split test learned from H0 in Figure 5. In the Figure, a number described subsequent to an attribute in parentheses represents a patient ID, and a leaf node predicts its majority class. A horizontal dashed line in a graph represents a border value of two categories (e.g. normal and high) of the corresponding medical test.

The decision tree and the time sequences employed in the Figure have attracted interests of physicians, and were recognized as an important discovery. The time-series tree investigates potential capacity of a liver by CHE and predicts patients with low capacity as liver cirrhosis. Then the tree investigates the degree of inflammation for other patients by GPT, and predicts patients with heavy inflammation as liver cirrhosis. For other patients, the tree investigates another sort of potential capacity of a liver by ALB and predicts liver cirrhosis based on the capacity. This procedure highly agrees with routine interpretation of blood tests by physicians. Our proposed method was highly appreciated by them since it discovered results which

Table 3. Experimental results with H0 using leave-one-out

method	accuracy (%)	time (s)	size
SE-split	88.2	0.5	6.9
Random 1	74.5	0.1	8.4
Random 10	78.4	0.3	7.4
Geurts	84.3	12.5	7.4
Kadous	78.4	1.6	5.1
CE-spit	62.7	0.5	8.5
Av-split	82.4	0.0	10.1
1-NN	74.5	0.1	N/A

are highly consistent to knowledge of physicians by only using medical knowledge on relevant attributes. They consider that we need to verify plausibility of this classifier in terms of as much information as possible from various sources then eventually move to biological tests.

During the quest, there was an interesting debate among the authors. Yamada and Suzuki, as machine learning researchers, proposed abstracting the time sequences in Figure 5. Yokoi and Takabayashi, who are physicians, however, insisted to use time sequences that exist in the data set[11]. The physicians are afraid that such abstracted time sequences are meaningless, and claim that the use of real time sequences is appropriate from the medical point of view.

We show the results of the learning methods with H0 in Table 3. Our time-series tree outperforms other methods in accuracy[12], and in tree size except for (Kadous, 1999). Test of significance based on two-tailed t-distribution shows that the differences of tree sizes are statistically significant. We can safely conclude that our method outperforms other methods when both accuracy and tree sizes are considered. Moreover, inspection of mis-predicted patients in leave-one-out revealed that most of them can be considered as exceptions. This shows that our method is also effective in detecting exceptional patients.

5. Conclusions

In applying a machine learning algorithm to real data, one often encounters the problems of quantity, qual-

[11]Other physicians supported Yokoi and Takabayashi. Anyway they are not reluctant to see the results of abstracted patterns too.

[12]By a binary test with correspondence, however, we cannot reject, for example, the hypothesis that SE-split and Av-split differ in accuracy. We need more examples to show superiority of our approach in terms of accuracy.

ity, and form. Data are massive (quantity), noisy (quality), and in various shapes (form). Recently, the third problem has motivated several researchers to propose classification from structured data including time-series, and we believe that this tendency will continue in the near future.

Information technology in medicine has spread its target to multimedia data such as time-series data and image data from string data and numerical data in the last decade, and aims to support the whole process of medicine which handles various types of data (Tanaka, 2001). Our decision-tree induction method which handles the shape of a time sequence explicitly, has succeeded in discovering results which were highly appreciated by physicians. We anticipate that there is a long way toward an effective classification method which handles various structures of multimedia data explicitly, but our method can be regarded as an important step toward this objective.

Acknowledgement

This work was partially supported by the grant-in-aid for scientific research on priority area "Active Mining" from the Japanese Ministry of Education, Culture, Sports, Science and Technology. We are grateful to Eamonn Keogh who permitted our use of his figures in this paper.

References

Almuallim, H., Akiba, Y., & Kaneda, S. (1996). An Efficient Algorithm for Finding Optimal Gain-ratio Multiple-split Tests on Hierarchical Attributes in Decision Tree Learning. *Proceedings of the Thirteenth National Conference on Artificial Intelligence* (pp. 703–708). Menlo Park, CA: AAAI Press.

Berka, P. (2002). ECML/PKDD 2002 Discovery Challenge, Download Data about Hepatitis. *http://lisp.vse.cz/challenge/ecmlpkdd2002/*.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and Regression Trees*. Belmont, CA: Chapman & Hall.

Drücker, C. et al. (2002). "As Time Goes by" - Using Time Series Based Decision Tree Induction to Analyze the Behaviour of Opponent Players. *RoboCup 2001: Robot Soccer World Cup V, LNAI 2377* (pp. 325–330). Berlin: Springer-Verlag.

Geurts, P. (2001). Pattern extraction for time series classification. *Principles of Data Mining and Knowledge Discovery, LNAI 2168* (pp. 115–127). Berlin: Springer-Verlag.

Hettich, S. and Bay, S. D. (1999). The UCI KDD Archive. *http://kdd.ics.uci.edu*. Irvine, CA: University of California, Department of Information and Computer Science.

Kadous, M. W. (1999). Learning comprehensible descriptions of multivariate time series. *Proceedings of the Sixteenth International Conference on Machine Learning* (pp. 454–463). San Francisco: Morgan Kaufmann.

Keogh, E. J. and Pazzani, M. J. (2000). Scaling up Dynamic Time Warping for Datamining Application. *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 285–289). New York: ACM.

Keogh, E. J. (2001). "Mining and Indexing Time Series Data". *Tutorial at the 2001 IEEE International Conference on Data Mining*. http://www.cs.ucr.edu/%7Eeamonn/tutorial_on_time_series.ppt.

Keogh, E. (2002). Exact Indexing of Dynamic Time Warping. *Proceedings of the 28th International Conference on Very Large Data Bases* (pp. 406–417). St. Louis: Morgan Kaufmann.

Mingers, J. (1989). An Empirical Comparison of Pruning Methods for Decision Tree Induction. *Machine Learning*, *4*, 227–243.

Murthy, S. K. (1998). Automatic Construction of Decision Trees from Data: A Multi-disciplinary Survey. *Data Mining and Knowledge Discovery*, *2*, 345–389.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*, San Mateo, CA: Morgan Kaufmann.

Rodríguez, J. J., Alonso, C. J., and Bostrvm, H. (2000). Learning First Order Logic Time Series Classifiers, *Proceedings of the Work-in-Progress Track at the Tenth International Conference on Inductive Logic Programming*, (pp. 260–275).

Sakoe, H. and Chiba, S. (1978). Dynamic Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Transaction on Acoustics, Speech, and Signal Processing*, *ASSP-26*, 43–49.

Takechi, F. and Suzuki, E. (2002). Finding an Optimal Gain-ratio Subset-split Test for a Set-valued Attribute in Decision Tree Induction. *Proceedings of the Nineteenth International Conference on Machine Learning* (pp. 618–625). San Francisco: Morgan Kaufmann.

Tanaka, H. (2001). *Electronic Patient Record and IT Medical Treatment*, Tokyo: MED (in Japanese).