Attribute selection based on a new conditional entropy for incomplete decision systems

Jianhua Dai *, Wentao Wang, Haowei Tian, Liang Liu

College of Computer Science, Zhejiang University, Hangzhou 310027, China

Abstract

Shannon’s entropy and its variants have been applied to measure uncertainty in rough set theory from the viewpoint of information theory. However, few studies have been done on attribute selection in incomplete decision systems based on information-theoretical measurement of attribute importance. In this paper, we introduce a new form of conditional entropy to measure the importance of attributes in incomplete decision systems. Based on the introduced conditional entropy, we construct three attribute selection approaches, including an exhaustive search strategy approach, a greedy (heuristic) search strategy approach and a probabilistic search approach for incomplete decision systems. To test the effectiveness of these methods, experiments on several real-life incomplete data sets are conducted. The results indicate that two of these methods are effective for attribute selection in incomplete decision systems.

1. Introduction

Rough set theory, introduced by Pawlak, is a useful mathematic approach for dealing with vague and uncertain information [1,2]. Rough set theory deals with the approximations of an arbitrary subset of a universe by two definable or observable subsets called lower and upper approximations. It has extracted much attention from researchers [3–16]. Attribute reduction or attribute selection is one of the most important topics of rough set theory [17–21]. However, classical rough set theory approaches are unsuitable for handling incomplete information systems or incomplete decision systems.

Recently, some extensions of classical rough set theory have been developed to handle incomplete information systems. They do not require any preprocessing, and do not require changing the size of the original system. Without loss of generality, missing values may have two possible explanations: “do not care” and “lost”. For “do not care” case, a tolerance relation, i.e. reflexive and symmetric but not transitive relation, was proposed [22,23]. For “lost” case, a similarity relation, i.e. reflexive and transitive but not symmetric relation, was proposed [24,25]. Based on the tolerance relation and discernibility function, an approach for the reduction of attributes was proposed in [22,23]. A later paper proposed the concept of a maximal consistent block and formulated a new, more accurate method of approximating object subsets in incomplete information systems [16]. Recently, Dai and Xu [26] also studied the extended uncertainty measurement issue in incomplete information systems by pure rough set approach. The entropy of a system, as proposed by Shannon [27] gives a measure of uncertainty about its actual structure. Information theory has been used in data processing widely [28–30]. Several authors have used Shannon’s entropy and its variants to measure uncertainty in rough set theory [17,31,32]. However, most of the studies focused on complete information system or complete decision system. Liang et al. [33] defined some new information entropies like rough entropy, combination entropy and combination granulation, which can be used in incomplete information system. Qian and Liang defined the combination entropy and combination granulation in incomplete information system, and gave some of their properties [34]. Qian et al. also proposed the conditional combination entropy, mutual information and defined a variant of combination entropy with maximal consistent block [35]. Considering decision attribute, Dai et al. [14] studied information entropies, including Shannons entropy, conditional entropy, and joint entropy for incomplete decision systems. Until recently, there is some lack of studies on attribute reduction for incomplete decision system based on information theory. The purpose of this paper is to construct a new information-theoretical measure of attribute importance in incomplete decision systems, and construct several attribute selection approaches based on this measure.

The remainder of this paper is organized as follows. Some preliminaries about rough set theory are reviewed in Section 2. In Section 3, a new form of conditional entropy for incomplete decision systems is introduced. Based on the conditional entropy, three attribute selection approaches are constructed in Section 4. The
proposed algorithms are tested on several real-life incomplete data sets in Section 5. Finally, we discuss and conclude our work in Section 6.

2. Preliminaries

In this section, we first review some basic concepts in rough set theory, which can also be referred to [1,2]. Furthermore, some preliminaries about incomplete information systems and the extended rough set models for incomplete information systems will be reviewed [22,23].

2.1. Information systems and decision systems

An information system is a quadruple $IS = (U, A, V, f)$, where: $U$ is a non-empty finite set of objects; $A$ is a non-empty finite set of attributes; $V$ is the value domain of attributes; $f$ is an information function which assigns particular values from domains of attributes to objects, such as $\forall a \in A, x \in U, f(a, x) \in V$, where $f(a, x)$ denotes the value of attribute $a$ on object $x$.

If there exist $a \in A$ and $x \in U$ such that $f(a, x)$ is equal to a missing value (a null or unknown value, denoted as ‘‘?’’), then the information system is called an incomplete information system (IIS). Otherwise, it is called a complete information system. Thus, an IFS can be expressed as:

$IIS = (U, A, V, f), \quad \star \in V$

A decision system is a quadruple $DS = (U, C \cup \{d\}, V, f)$, where $d$ is the decision attribute, $C$ is the conditional attribute set. If there exist $a \in C$ and $x \in U$ such that $f(a, x)$ is equal to a missing value (a null or unknown value, denoted as ‘‘?’’), then the decision system is called an incomplete decision system (IDS). Thus, an IDS can be expressed as:

$IDS = (U, A, C \cup \{d\}, V, f), \quad \star \in V_C$

Given an incomplete decision system $IDS = (U, C \cup \{d\}, V, f), \quad \star \in V_C$, for any subset of condition attributes $B \subseteq C$, a tolerance relation $T$ can be defined as [22]:

$T_B = \{(x, y) | \forall a \in B, f(a, x) = f(a, y) \vee f(a, x) = \star \vee f(a, y) = \star\}$

Obviously, $T$ is reflexive and symmetric, but may not be transitive. The tolerance class of an object $x$ with respect to an attribute set $B$ is defined as:

$T_B(x) = \{y \in U | (x, y) \in T_B\}$

Definition 1. Let $IDS = (U, A, C \cup \{d\}, V, f), \quad \star \in V_C$ be an incomplete decision system, then the generalized decision $\partial_B: U \rightarrow V_D$ is defined as follows:

$\partial_B(x) = \{i \in \bar{i} \mid f(d, y) \wedge y \in T_B(x)\}$

Definition 2. Let $IDS = (U, A, C \cup \{d\}, V, f), \quad \star \in V_C$ be an incomplete decision system. Let $\partial_B: U \rightarrow V_D$ be the generalized decision function. If $\partial_B(x) = 1$ for any $x \in U$, then the incomplete decision is consistent (deterministic, definite), where $| \cdot |$ denotes the number of elements of the set.

Example 1. Given an incomplete decision system shown in Table 1, where $U = \{u_1, u_2, u_3, u_4, u_5, u_6\}, C = \{\text{Price, Mileage, Size, Max - Speed}\} = \{P, M, S, X\}$ and $d = \text{Acceleration}$.

<table>
<thead>
<tr>
<th>Car</th>
<th>Price</th>
<th>Mileage</th>
<th>Size</th>
<th>Max-speed</th>
<th>Acceleration</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>High</td>
<td>High</td>
<td>Full</td>
<td>Low</td>
<td>Good</td>
</tr>
<tr>
<td>$u_2$</td>
<td>High</td>
<td>*</td>
<td>Full</td>
<td>High</td>
<td>Good</td>
</tr>
<tr>
<td>$u_3$</td>
<td>*</td>
<td>*</td>
<td>Compact</td>
<td>High</td>
<td>Poor</td>
</tr>
<tr>
<td>$u_4$</td>
<td>*</td>
<td>*</td>
<td>Full</td>
<td>Low</td>
<td>Good</td>
</tr>
<tr>
<td>$u_5$</td>
<td>Low</td>
<td>*</td>
<td>Full</td>
<td>High</td>
<td>Excellent</td>
</tr>
<tr>
<td>$u_6$</td>
<td>*</td>
<td>High</td>
<td>Full</td>
<td>Low</td>
<td>Good</td>
</tr>
</tbody>
</table>

Thus, we get $|\partial_C(u_1)| = |\partial_C(u_2)| = |\partial_C(u_3)| = |\partial_C(u_4)| = |\partial_C(u_5)| = |\partial_C(u_6)| = 1$. Hence, the incomplete decision system is consistent.

Definition 3. Let $IDS = (U, C \cup \{d\})$ be an incomplete decision system. Let $B \subseteq C$, then the attribute set $B$ is a relative redact of $IDS$ if and only if

(1) $\partial_B(x) = \partial_C(x)$, for all $x \in U$.
(2) $\forall b \in B, \partial_{B \setminus \{b\}} \neq \partial_C$.

3. Conditional entropy measure for incomplete decision system

In this section, we will use a new form of conditional entropy in incomplete decision system based on the tolerance relation based on the work of Dai et al. [32]. Some properties of the conditional entropy will be given.

Definition 4. Given a consistent incomplete decision system $IDS = (U, C \cup \{d\})$ and $B \subseteq C$. Let $U/T_B = \{T_B(x_1), T_B(x_2), \ldots, T_B(x_U)\}$, $U/d = D_1, D_2, \ldots, D_m$. The conditional entropy of $B$ to $d$ is defined as follows:

$H(d|B) = - \sum_{i=1}^{m} p(T_B(u_i)) \sum_{j=1}^{U/d} p(D_j|T_B(u_i)) \log p(D_j|T_B(u_i))$ (1)

where

$p(T_B(u_i)) = \frac{|T_B(u_i)|}{|U|}, \quad i = 1, 2, \ldots, |U|$

$p(D_j|T_B(u_i)) = \frac{|T_B(u_i) \cap D_j|}{|T_B(u_i)|}, \quad i = 1, 2, \ldots, |U|, \quad j = 1, 2, \ldots, m$

It is worth noting that we define $H(d|B) = 0$ when $T_B(x_i) \cap D_j = \emptyset$.

Proposition 1. Let $IDS = (U, C \cup \{d\})$ be a consistent incomplete decision system. Then, we have $H(d|C) = 0$.

Proof. Since IDS is consistent, we have $\partial_B(x_i) = 1$, $\forall x_i \in U$. It means $T_C(x_i) \subseteq D_j$, $D_j \in U/d$, $\forall x_i \in U$

Hence, we have

$T_C(x_i) \cap D_j = \emptyset, \quad D_j \in U/d, \quad D_j \neq D_j$

Consequently, we have

$\sum_{j=1}^{m} \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} = 0 + 0 + \cdots + 1 \log 1 + 0 + \cdots + 0 = 0$

Thus, we know
H(d|C) = \sum_{j=1}^{[U]} p(T_B(u_j)) \sum_{j=1}^{[U]} \frac{|T_B(x_j) \cap D_j|}{|T_B(x_j)|} = 0 \quad \square

Proposition 2. Let IDS = (U, C \cup \{d\}) be a consistent incomplete decision system, \( B \subseteq C \) is a relative reduct of \( C \) relative to decision attribute \( d \), and only if

1. \( H(d|C) = H(d|B) \)
2. \( \forall B \subseteq B, H(d|B') \neq H(d|B) \)

Proof. (a) We first prove that if \( H(d|C) = H(d|B) \), \( \forall B \subseteq B, H(d|B') \neq H(d|B) \) then \( B \subseteq C \) is a relative reduct.

By Proposition 1, we know \( H(d|C) = 0 \). Then \( H(d|B) = H(d|C) = 0 \).

Since \( 0 < \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \leq 1 \) and \( 0 < \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \leq 1 \), we have

\[
\frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \log \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \leq 0
\]

At the same time, we know \( H(d|B) = 0 \) if and only if

\[
\frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \log \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} = 0, \quad \forall x_i, D_j
\]

It means that there exists only one \( D_j \in U/d \) such that \( T_B(x_i) \subseteq D_j \) and \( g(x_i) \subseteq D_j = \emptyset \). Thus, \( H(d|B) = H(d|C) \).

Since \( B \subseteq C \) is a relative reduct, we get \( \partial_B = \partial_C \). i.e. \( \partial_B(x_i) = \partial_C(x_i) = 1 \), \( \forall x_i \in U \). Hence, \( \partial_B \neq \partial_B \).

(b) We then prove that if \( B \subseteq C \) is a relative reduct, then \( H(d|C) = H(d|B) \).

Since \( B \subseteq C \) is a relative reduct, we get \( \partial_B = \partial_C \). i.e. \( \partial_B(x_i) = \partial_C(x_i) = 1 \), \( \forall x_i \in U \). Hence, \( \partial_B \neq \partial_B \).

It means that \( \partial_B(x_i) = 1 \). Hence, \( \partial_B \neq \partial_B \).

It follows that there exist \( x_i \), \( D_j \) such that

\[
T_B(x_i) \subseteq D_j
\]

and

\[
T_B(x_i) \subseteq D_j = \emptyset, \quad \forall x_i, D_j
\]

It follows that

\[
\frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \log \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} = 0, \quad \forall x_i, D_j
\]

It means

\[
\sum_{j=1}^{[U]} \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} \log \frac{|T_B(x_i) \cap D_j|}{|T_B(x_i)|} = 0, \quad \forall x_i
\]

Hence, we know \( H(d|B) = 0 \).

4. Attribute selection approaches for incomplete decision systems

Attribute selection methods contain two important aspects: evaluation of a candidate attribute subset and search strategy through the attribute space [36]. We use the conditional entropy discussed in Section 3 to evaluate the attribute subset and a measure is defined as follows:

Definition 5. Given a consistent decision system IDS = (U, C \cup \{d\}), let \( B \subseteq C \) and \( a \in C - B \). Then the significance of attribute \( a \) relative to \( B \) is defined as

\[
\Sigma(a, B, d) = H(d|B) - H(d|B \cup \{a\})
\]

This definition describes the increment of discernibility power relative to the decision caused by introducing attribute \( a \). Thus, it can be used as a measure for attribute selection in incomplete decision system. Based on the above measure, three attribute selection approaches based on different search strategies are proposed, which are exhaustive-Focus [37], heuristic-SetCover [38] and probabilistic-Las Vegas Filter (LVF) [39].

4.1. inFocus: exhaustive search

Focus is one of earliest attribute selection or feature selection algorithms in machine learning [36,37] and it begins with an empty attribute set and carries out breadth-first search until it finds a minimal subset that satisfies stop criterion. Since Focus uses an exhaustive search strategy, it guarantees an optimal solution. Here, we give the Focus approach for incomplete decision system as follows, called inFocus.

Algorithm 1. inFocus

Input: An incomplete decision system \((U, C \cup \{d\})\).
Output: An attribute selection result \( R \).
1. For every size = 0 to \( |C| \)
2. For all subset SelectAttr with \(|\text{SelectAttr}| = \text{size}\)
3. If \( H(d|\text{SelectAttr}) \neq H(d|C) \)
4. \( R = \text{SelectAttr} \)

Example 2 (Continued from Example 1). Given the incomplete decision system shown in Table 1, we have:

- \( C = \{P, M, S, X\} \)
- \( U/d = \{\{u_1, u_2, u_4, u_6\}, \{u_7\}\} \)
- \( T_P(u_1) = T_P(u_2) = \{u_1, u_2, u_3, u_4, u_6\} \)
- \( T_P(u_3) = T_P(u_4) = T_P(u_5) = \{u_1, u_2, u_3, u_4, u_5, u_6\} \)
- \( T_P(u_7) = \{u_3, u_4, u_5, u_6\} \)

\[
H(d|P) = - \left[ 2 \cdot \frac{5}{6} \log \frac{5}{6} + \frac{4}{5} \log \frac{4}{5} + \frac{1}{5} \log \frac{1}{5} \right] + 3 \cdot \frac{6}{6} \log \frac{6}{6} + \frac{2}{6} \log \frac{2}{6} + \frac{1}{6} \log \frac{1}{6} \]

\[
+ 4 \cdot \frac{2}{4} \log \frac{2}{4} + \frac{2}{4} \log \frac{4}{4} \right] = 5.9581
\]

\[
T_M(u_1) = T_M(u_2) = T_M(u_3) = T_M(u_4) = T_M(u_5) = T_M(u_6)
\]

\[
= \{u_1, u_2, u_3, u_4, u_6\}
\]

\[
H(d|M) = - \left[ 6 \cdot \frac{6}{6} \log \frac{6}{6} + \frac{4}{6} \log \frac{4}{6} + \frac{2}{6} \log \frac{2}{6} + \frac{1}{6} \log \frac{1}{6} \right] = 7.5098
\]
The processing of Focus approach.

Fig. 1. The processing of Focus approach.
4.2. inSetCover: heuristic search

We can also use heuristic search or greed search to find attribute reduction. At the beginning, the candidate attribute subset is empty. Then, a new attribute which maximizes the significance measure is added to the selected attribute subset each time, until the stop criterion is satisfied. SetCover is fast, close to optimal, and deterministic [36,38]. Here, we give the SetCover approach for incomplete decision system as follows, called inSetCover.

Algorithm 2. inSetCover

Input: An incomplete decision system \((U, C \cup \{d\})\).
Output: Selected attribute subset \(R\).
1. Step 1. Initialize:
   \(\text{SelectAttr} = \emptyset\)
   \(\text{unSelectAttr} = C\)
2. Step 2. Select attribute:
   \(\text{(a)}\) For every attribute \(a_i \in \text{unSelectAttr}\
   \text{Attr} = \text{SelectAttr} \cup \{a_i\},\) calculate the tolerance class of object \(x_p, T_{\text{Attr}}(x_p)\).
   \(\text{(b)}\) Calculate the conditional entropy \(H(d|\{a_i\})\)
   \(\text{(c)}\) choose the attribute \(a\) which minimizes \(H(d|\text{Attr} \cup \{a\})\), i.e. choose the attribute with the maximum significance measure \(\text{Sig}(a, \text{SelectAttr}, d)\);
3. Step 3. SelectAttr = SelectAttr \cup \{a\}.
   unSelectAttr = unSelectAttr \{-a\}.
4. Step 4. If \(H(d|\text{Attr}) \neq H(d|C)\), go to Step 2, otherwise \(R = \text{SelectAttr}\).

Example 3 (Continued from Example 1). Given the incomplete decision system shown in Table 1, we have

1. SelectAttr = \emptyset, SelectAttr = \(C = \{P, M, S, X\}\) and \(U|d = \{(u_1, u_2, u_4, u_6)\}\).
2. \(H(d|\{P\}) = -\left[2 * \frac{5}{6} * \left(\frac{4}{9} \log_2 \frac{4}{9} + \frac{1}{9} \log_2 \frac{1}{9}\right) + 3 * \frac{6}{6} * \left(\frac{4}{6} \log_2 \frac{4}{6} + \frac{2}{6} \log_2 \frac{2}{6}\right) + 4 * \frac{4}{6} * \left(\frac{2}{4} \log_2 \frac{2}{4} + \frac{2}{4} \log_2 \frac{2}{4}\right)\right] = 5.9581\)
3. \(H(d|M) = -\left[6 * \frac{6}{6} * \left(\frac{4}{6} \log_2 \frac{4}{6} + \frac{2}{6} \log_2 \frac{2}{6}\right)\right] = 7.5098\)
4. \(H(d|S) = -\left[5 * \frac{5}{6} * \left(\frac{4}{5} \log_2 \frac{4}{5} + \frac{1}{5} \log_2 \frac{1}{5}\right)\right] = 3.0080\)
5. \(H(d|X) = -\left[3 * \frac{3}{6} * 0 + 3 * \frac{3}{6} \log_2 \frac{1}{3}\right] = 2.3774\)

Since \(H(d|X)\) is the minimum one, we choose the attribute \(X\). Thus SelectAttr = SelectAttr \cup \{X\} = \{X\}, and \(\text{unSelectAttr} = \text{unSelectAttr} \{-X\} = \{P, M, S\}\). Since \(H(d|\text{SelectAttr}) \neq 0\), we need to choose more attributes.

3. \(H(d|\{P, X\}) = -\left[2 * \frac{2}{6} * \left(\frac{1}{2} \log_2 \frac{1}{2} + \frac{1}{2} \log_2 \frac{1}{2}\right) + 3 * \frac{3}{6} * \left(\frac{1}{2} \log_2 \frac{1}{3} + \frac{1}{2} \log_2 \frac{1}{3}\right)\right] = 1.4591\)
4. \(H(d|M, X) = -\left[3 * \frac{3}{6} * \left(\frac{1}{3} \log_2 \frac{1}{3} + \frac{1}{3} \log_2 \frac{1}{3} + \frac{1}{3} \log_2 \frac{1}{3}\right)\right] = 2.3774\)

Since \(H(d|\{S, X\})\) is the minimum one, we choose the attribute \(S\). Thus SelectAttr = SelectAttr \cup \{S\} = \{X, S\}, and \(\text{unSelectAttr} = \text{unSelectAttr} \{-S\} = \{P, M\}\). Since \(H(d|\text{SelectAttr}) \neq 0\), we need to choose more attributes.

4.3. inLVF: probabilistic search

Las Vegas algorithms [40] for feature subset selection can make probabilistic choices of subsets in search of an optimal set. Las Vegas Filter (LVF) is a probabilistic algorithm where probabilities of generating any subset are equal [39]. In this paper, we use the introduced conditional entropy as LVF’s evaluation measure. It generates attribute subsets randomly with equal probability, and records the minimal size of attributes subset satisfying the stop criterion of Maximum Tries times. LVF is fast in reducing the number of features in the early stages and can produce optimal solutions if computing resources permit [36]. The LVF approach for attribute selection in incomplete decision systems is given as follows, called inLVF.

Algorithm 3. inLVF

Input: An incomplete decision system \((U, C \cup \{d\}), \text{MaxTries}\).
Output: Selected attribute subset \(R\).
1. SelectAttr = \(C\)
2. For \(j = 0\) to \(\text{MaxTries}\)
3. Randomly choose a subset of attribute \(S_j\)
4. If \(H(d|S_j) \neq H(d|C)\) \&\& \(|S_j| < |\text{SelectAttr}|\)
5. SelectAttr = \(S_j\)
6. \(R = \text{SelectAttr}\)
5. Experiments

Several real-life incomplete data sets have been used in our experiments. These data sets are available from the UCI Repository of Machine Learning Database at University of California [41]. The summary of the data sets are summarized in Table 2.

In Focus algorithm, we terminate the program when it runs beyond 5000 s. We set inLVF algorithm’s MaxTries be 1000.

The size of selected attribute subset and running time of each attribute selection approach are shown in Table 3. From Table 3, we find that Focus approach takes much more time to obtain an attribute reduction comparing with the other two approaches. Three

*Table 2*
Summary of the experimental datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Objects</th>
<th>Features</th>
<th>Classes</th>
<th>Incomplete Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrhythmia</td>
<td>452</td>
<td>279</td>
<td>0</td>
<td>279</td>
</tr>
<tr>
<td>Audiology</td>
<td>226</td>
<td>69</td>
<td>69</td>
<td>0</td>
</tr>
<tr>
<td>Cleveland</td>
<td>303</td>
<td>13</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>Colic</td>
<td>368</td>
<td>22</td>
<td>15</td>
<td>7</td>
</tr>
<tr>
<td>Credit</td>
<td>690</td>
<td>15</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>155</td>
<td>19</td>
<td>13</td>
<td>6</td>
</tr>
<tr>
<td>Labor</td>
<td>57</td>
<td>16</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

*Table 3*
Size of selected attribute subset and running time.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Focus</th>
<th>SetCover</th>
<th>LVF</th>
<th>FullFeatures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrhythmia</td>
<td>–</td>
<td>5000</td>
<td>5000</td>
<td>5000</td>
</tr>
<tr>
<td>Audiology</td>
<td>–</td>
<td>5000</td>
<td>5000</td>
<td>5000</td>
</tr>
<tr>
<td>Cleveland</td>
<td>10</td>
<td>68.58</td>
<td>10</td>
<td>68.58</td>
</tr>
<tr>
<td>Colic</td>
<td>–</td>
<td>5000</td>
<td>18</td>
<td>23.4</td>
</tr>
<tr>
<td>Credit</td>
<td>13</td>
<td>1037.14</td>
<td>13</td>
<td>2.94</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>14</td>
<td>1072.40</td>
<td>14</td>
<td>0.31</td>
</tr>
<tr>
<td>Labor</td>
<td>10</td>
<td>20.97</td>
<td>10</td>
<td>0.03</td>
</tr>
<tr>
<td>Average</td>
<td>11.75</td>
<td>549.77</td>
<td>13.6</td>
<td>23.26</td>
</tr>
</tbody>
</table>

*Table 4*
Performance of the attribute selection approaches with RBF-SVM classifier.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Focus</th>
<th>SetCover</th>
<th>LVF</th>
<th>CFS</th>
<th>FullAttributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrhythmia</td>
<td>–</td>
<td>54.2 ± 0.0</td>
<td>54.2 ± 0.0</td>
<td>54.2 ± 0.0</td>
<td>54.2 ± 0.0</td>
</tr>
<tr>
<td>Audiology</td>
<td>–</td>
<td>64.85 ± 0.55</td>
<td>53.32 ± 0.89</td>
<td>66.9 ± 0.77</td>
<td>53.32 ± 0.89</td>
</tr>
<tr>
<td>Cleveland</td>
<td>55.54 ± 0.96</td>
<td>55.4 ± 0.96</td>
<td>53.1 ± 0.69</td>
<td>55.31 ± 0.83</td>
<td>54.13 ± 0.0</td>
</tr>
<tr>
<td>Colic</td>
<td>–</td>
<td>69.48 ± 1.04</td>
<td>66.06 ± 1.03</td>
<td>68.64 ± 0.75</td>
<td>66.06 ± 1.03</td>
</tr>
<tr>
<td>Credit</td>
<td>56.13 ± 0.26</td>
<td>55.3 ± 0.21</td>
<td>55.81 ± 0.25</td>
<td>56.45 ± 0.39</td>
<td>55.81 ± 0.25</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>82.32 ± 0.45</td>
<td>82.32 ± 0.45</td>
<td>79.35 ± 0.0</td>
<td>78.9 ± 0.81</td>
<td>79.35 ± 0.0</td>
</tr>
<tr>
<td>Labor</td>
<td>91.75 ± 1.44</td>
<td>91.75 ± 1.44</td>
<td>94.21 ± 0.85</td>
<td>92.28 ± 0.91</td>
<td>94.21 ± 0.85</td>
</tr>
<tr>
<td>Average(Acc)</td>
<td>–</td>
<td>68.15</td>
<td>65.15</td>
<td>67.53</td>
<td>65.30</td>
</tr>
</tbody>
</table>

*Table 5*
Performance of the attribute selection approaches with C4.5 classifier.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Focus</th>
<th>SetCover</th>
<th>LVF</th>
<th>CFS</th>
<th>FullAttributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrhythmia</td>
<td>–</td>
<td>62.21 ± 1.72</td>
<td>65.46 ± 1.47</td>
<td>69.58 ± 0.85</td>
<td>65.46 ± 1.47</td>
</tr>
<tr>
<td>Audiology</td>
<td>–</td>
<td>75.66 ± 0.55</td>
<td>77.43 ± 0.78</td>
<td>77.74 ± 0.91</td>
<td>77.43 ± 0.78</td>
</tr>
<tr>
<td>Cleveland</td>
<td>54.09 ± 1.68</td>
<td>54.09 ± 1.68</td>
<td>54.3 ± 2.0</td>
<td>54.39 ± 2.03</td>
<td>52.05 ± 1.56</td>
</tr>
<tr>
<td>Colic</td>
<td>–</td>
<td>85.33 ± 0.38</td>
<td>85.24 ± 0.29</td>
<td>80.63 ± 0.75</td>
<td>85.24 ± 0.29</td>
</tr>
<tr>
<td>Credit</td>
<td>85.72 ± 0.49</td>
<td>85.68 ± 0.58</td>
<td>85.74 ± 0.5</td>
<td>84.41 ± 0.71</td>
<td>85.74 ± 0.5</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>80.00 ± 1.49</td>
<td>80.0 ± 1.49</td>
<td>79.42 ± 2.31</td>
<td>80.19 ± 1.77</td>
<td>79.42 ± 2.31</td>
</tr>
<tr>
<td>Labor</td>
<td>79.12 ± 3.83</td>
<td>79.12 ± 3.83</td>
<td>78.42 ± 2.99</td>
<td>79.65 ± 4.07</td>
<td>78.42 ± 2.99</td>
</tr>
<tr>
<td>Average(Acc)</td>
<td>–</td>
<td>75.81</td>
<td>75.55</td>
<td>75.23</td>
<td>74.82</td>
</tr>
</tbody>
</table>

*Table 6*
Performance of the algorithms with CART classifier.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Focus</th>
<th>SetCover</th>
<th>LVF</th>
<th>CFS</th>
<th>FullAttributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrhythmia</td>
<td>–</td>
<td>70.93 ± 0.72</td>
<td>71.13 ± 0.9</td>
<td>72.12 ± 0.76</td>
<td>71.13 ± 0.9</td>
</tr>
<tr>
<td>Audiology</td>
<td>–</td>
<td>72.79 ± 0.98</td>
<td>74.12 ± 1.29</td>
<td>77.26 ± 1.22</td>
<td>74.12 ± 1.29</td>
</tr>
<tr>
<td>Cleveland</td>
<td>55.68 ± 0.82</td>
<td>55.68 ± 0.82</td>
<td>55.61 ± 0.65</td>
<td>55.48 ± 0.99</td>
<td>55.38 ± 1.26</td>
</tr>
<tr>
<td>Colic</td>
<td>–</td>
<td>85.41 ± 0.64</td>
<td>85.41 ± 0.64</td>
<td>80.65 ± 0.79</td>
<td>85.41 ± 0.64</td>
</tr>
<tr>
<td>Credit</td>
<td>85.00 ± 0.46</td>
<td>84.99 ± 0.42</td>
<td>84.99 ± 0.42</td>
<td>84.97 ± 0.31</td>
<td>84.99 ± 0.42</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>79.10 ± 1.78</td>
<td>79.1 ± 1.78</td>
<td>77.23 ± 1.67</td>
<td>78.39 ± 1.73</td>
<td>77.23 ± 1.67</td>
</tr>
<tr>
<td>Labor</td>
<td>81.75 ± 3.98</td>
<td>81.75 ± 3.98</td>
<td>80.35 ± 2.59</td>
<td>81.58 ± 4.07</td>
<td>80.35 ± 2.59</td>
</tr>
<tr>
<td>Average(Acc)</td>
<td>–</td>
<td>75.81</td>
<td>75.55</td>
<td>75.78</td>
<td>75.52</td>
</tr>
</tbody>
</table>
data sets of seven are not computed to obtain results in our preset time 5000 s. It indicates that Focus approach is really time consuming. Table 3 also shows that SetCover approach tends to select fewer attributes comparing with LVF approach. In other words, the size of attribute subset selected by SetCover approach intends to be smaller than the size of attribute subset selected by LVF approach.

To evaluate these attribute selection approaches, we employ three learning mechanisms to create classifiers. All these selected algorithms are from or implemented in Weka [42]. The average accuracy and standard deviation of 10 run times with 10-fold cross-validation of RBF-SVM, C4.5, CART are shown in Tables 4–6 respectively. It should be noticed that the commonly used feature selection method CFS [43] and Full attributes join the comparison.

From Tables 4–6, one can see that the constructed attribute selection approaches inSetCover and inLVF based on the introduced conditional entropy for incomplete data are effective. After attribute selection, the classification accuracy is higher than full attributes before selection. Comparing with CFS, inSetCover also has a higher performance. We may conclude that the constructed approaches inSetCover and inLVF are both effective for attribute selection in incomplete data.

6. Conclusion

In this paper, we use a new form of conditional entropy to measure the importance of attributes in incomplete decision systems. Based on this measure, three types of attribute selection approaches for incomplete decision systems, including the exhaustive search strategy approach inFocus, the heuristic or greedy search strategy approach inSetCover and the probabilistic search approach inLVF are constructed. To test the effectiveness of the constructed attribute selection approaches, experiments on several real-life incomplete data sets are conducted. Results show that inSetCover and inLVF are effective. The numbers of attributes are reduced obviously. At the same time, the classification performances increase after the attribute reduction. We expect the introduced approaches could supply optional solutions for attribute selection in incomplete decision systems.

Acknowledgement

This work was partially supported by the National Natural Science Foundation of China (Nos. 61070074, 60703038).

References

[34] Y. Qian, J. Liang, Combination entropy and combination granulation in incomplete information system, LNAI 4062 (2006) 184–190.