Chapter 5

Discovery of Production Rules with Exceptions and Fuzzy Hierarchy

In this chapter, an algorithm for the Discovery of Production Rules with Exceptions and Fuzzy Hierarchy (O^PR^F3e) is presented as an extension of the O^PR^F3e-algorithm discovered in previous chapter 4. It is to be noted that under O^PR^F3e-algorithm weak classes could not become the part of the hierarchy whereas discovery of exceptions for weak classes under O^PR^F3e-algorithm results in the inclusion of weak classes as well in the fuzzy hierarchy. However, the weak classes having no exceptions would not appear in the fuzzy hierarchy and are termed as irrelevant classes.

5.1. Characteristics of the Exceptions in the Fuzzy Hierarchy

The exception (censor) conditions with respect to Censored Production Rules (CPRs) have been discussed in chapter 2, and we have seen that the exception conditions hold rarely and are covered by only a small portion of data.

The exception condition to the class D_k in the fuzzy hierarchical structure is the exception condition to each class in the sub-trees with D_k as the root i.e. if the exception condition to the class D_k happens to be true then all the classes in the sub-trees with D_k as the root (including class D_k) are blocked. For example the exception conditions “Penguin v Wings_broken” to the class Fly in Fig.5.1 are automatically exception conditions to all the classes in the fuzzy hierarchical structure with Fly as the root i.e. Flapping, Gliding, Soaring, Hovering, Thermal and Slop. The properties P_1, P_2, ..., P_8 in Fig.5.1 having the following meaning:

P_1: Bird beats its wings downward and forward.
P_2: Bird stretches its wings out.
P_3: Bird holds its wings.
P_4: Bird loses altitude.
P_5: Bird uses the air currents.
P_6: Bird uses rising warm air current to left them.
Fly[Bird] $(\text{Penguin} \lor \text{Wings\_broken})$

Flapping $[P_1]$  Gliding $[P_2, P_3, P_4]$  Soaring $[P_5]$ $(\text{American Crows} \lor \text{Hummingbird})$

Hovering $[P_6]$  Slop$[P_7]$

Fig. 5.1: Fuzzy Hierarchical Structure with exceptions for different types of bird flying.

5.2. Production Rules with Exceptions and Fuzzy Hierarchy

Intuitively, exceptions contradict the common sense rules i.e. General Rules (GR) and they have low support [65]. Therefore, if we mine exception rules (ER), literally there would be a huge number of them because of their usual lower support threshold. An exception rule is structurally defined in [11, 100] as follows:

GR: $P \rightarrow D_k.$

ER: $P \rightarrow D_k \bot E_k.$

i.e. $P \land E_k \rightarrow \sim D_k,$

where $D_k \land \sim D_k = \text{FALSE}.$ That is, $D_k$ and $\sim D_k$ logically contradict each other.

The set of premises $P$ in GR holds on a statistically large subset of tuples in $D_k.$ Since both GR and ER hold statistically, the intersection of their premises is very small.

In order to define the concept of Production Rules with Exceptions and Fuzzy Hierarchy, we have to distinguish between the contradict classes and specific classes with respect to the general rule GR.

Given two rules $P_a \rightarrow D_a$ and $P_b \rightarrow D_b$ such that $P_a \subset P_b,$ both the classes $D_a$ and $D_b$ logically contradict each other if

- $|P_a \land D_a| \gg |P_b \land D_b|,$ and
- $P_b \not\subset D_a.$
It is to be noted that the set of exception conditions, $E_k$, to the class $D_k$ in the fuzzy hierarchical structure is disjoint with the set of premises $P_{k+1}$ of the specific class $D_{k+1}$ i.e. $E_k \cap P_{k+1} = \emptyset$. Also the set of exception conditions, $E_k$, to the specific class $D_k$ in the fuzzy hierarchical structure is disjoint with the set of premises $P_{k-1}$ of the general class $D_{k-1}$ i.e. $E_k \cap P_{k-1} = \emptyset$ (see Fig.5.2).

\[
\begin{array}{c|c|c}
\text{Level } k-1 & D_{k-1} & E_{k-1} \\
& \left[ P_{k-1} \right] & \downarrow d_1 \\
\text{Level } k & D_k & E_k \\
& \left[ P_k \right] & \downarrow d_2 \\
\text{Level } k+1 & D_{k+1} & E_{k+1} \\
& \left[ P_{k+1} \right] & \downarrow \\
\end{array}
\]

Fig.5.2: Proposed Fuzzy Hierarchical structure with Exceptions.

5.3. Exception Mining using Freq Matrix

Exception mining is the process of trawling through Freq matrix in hope of identifying the interpretable exception to each weak class.

The exception conditions play very important role in allowing all/some of the weak classes to become part of the fuzzy hierarchy. A particular weak class $D_k$ ($1 \leq k \leq n$) can become part of the fuzzy hierarchy if

- There exists a set of exceptions, $E_k$, to the class $D_k$ discovered from the contradict class $D_b \ \forall \ b \neq k$,
- For a given general class $D_g$ of $D_k$, the degree of subsumption $\text{deg}_{\text{sub}}(D_g, D_k) \geq \text{Threshold}$ value,
- There exists highlighted elements in $D_k$ corresponding to the root properties $P_{\text{root}}$ i.e. class $D_k$ must have all the properties in $P_{\text{root}}$, and
- If all other classes inherit the properties of $D_k$ then these classes must not contradict $D_k$.

If the above conditions fail then the weak class $D_k$ is irrelevant class and would not become part of the fuzzy hierarchy. Removing such irrelevant classes will reduce the size of the Freq matrix.
For a given weak class $D_a$ in the Freq matrix, consider a subset $\alpha_a \subseteq (P_{pub})_a$. The property $P_j \notin \alpha_a$ in the class $D_b$ ($D_b \neq D_a$) is a single exception condition (i.e. $E_a = \{P_j\}$) to the class $D_a$ if the following conditions are satisfied:

(i) $\alpha_a$ belongs to both the classes $D_a$ (covering large portion of data) and $D_b$ (covering small portion of data). To ensure that $\alpha_a$ in the class $D_b$ occurs rarely,

$$|\alpha_a \land D_a| \gg |\alpha_a \land D_b| \quad \text{....(5.1)}$$

This condition can be verified using the Freq matrix without dataset access if $f_a > 0$ and $f_b > 0$ where,

$$f_a = (\sum_{i=1}^{\alpha_a} \text{Freq}[P_i, D_a]) - (|\alpha_a| - 1), \quad \forall \ P_i \in \alpha_a, \text{ and} \quad \text{....(5.2)}$$

$$f_b = (\sum_{i=1}^{\alpha_a} \text{Freq}[P_i, D_b]) - (|\alpha_a| - 1), \quad \forall \ P_i \in \alpha_a \quad \text{....(5.3)}$$

So,

$$f_a \ast |D_a| \gg f_b \ast |D_b| \quad \text{....(5.4)}$$

(ii) The exception condition $P_j$ is not associated with $\alpha_a$ in the class $D_a$ i.e. the combination $\alpha_a \land P_j$ is not belong to $D_a$ e.g. $\text{Freq}[P_j, D_a] = 0$.

(iii) The exception condition $P_j$ to the class $D_a$ must belong to class $D_b$ i.e. $\text{Freq}[P_j, D_b] > 0$.

(iv) $\alpha_a \land P_j$ belongs to the class $D_b$ such that $P_j$ covers all the instances $\alpha_a \land D_b$,

$$|\alpha_a \land D_b| = |\alpha_a \land P_j \land D_b| \quad \text{....(5.5)}$$

Without dataset access, using $f_b$ (formula (5.3)) and the Freq matrix we can ensure that condition (iv) is not satisfied if

$$\text{Freq}[P_j, D_b] < f_b \quad \text{....(5.6)}$$

The weak class $D_a$ in the Freq matrix may have multiple exception conditions (i.e. $|E_a| > 1$). The exception conditions may either belong to a single class $D_b \ \forall \ b \neq a$, or to different classes $D_b, D_c, \ldots, D_k, \ldots \ \forall \ k \neq a$ i.e. $E_a = E_b \cup E_c \cup \ldots \cup E_k \cup \ldots$. The multiple exception conditions to class $D_a$ exist if

(a) Condition (i) is true.

(b) Both the conditions (ii) and (iii) are true for each exception condition $P_j \in E_a$. 

87
(c) Each instance in \( \alpha_a \land D_b \) must be associated with at least one exception condition \( P_j \in E_a \).

Without dataset access, using \( f_b \) (formula (5.3)) and the Freq matrix we can ensure that condition (c) is not satisfied if

\[
|E^b_a| \sum_{j=1}^{\frac{|E^b_a|}{f_b}} \text{Freq}[P_j, D_b] < f_b, \quad b \neq a ....(5.7)
\]

Consider the situation in which \( \alpha_a \) covers large portion of data of the weak class \( D_a \) and it also covers large portion of data of other weak classes \( D_1, D_2, ..., D_k, ..., \forall k \neq a \). The specific and general relation between the weak classes is defined as:

(i) \( D_a \) is a general class of \( D_k \) if
- \( \alpha_a \in D_k \) such that \( \alpha_a \subset \alpha_k \),
- \( \alpha_k \in D_a \), and
- \( \exists \ P_j \in E_a \) such that \( P_j \land \alpha_k \in D_k \).

(ii) \( D_a \) is specific class of \( D_k \) if
- \( \alpha_a \in D_k \) where \( \alpha_k \subset \alpha_a \),
- \( \alpha_k \in D_a \), and
- \( P_j \not\in D_a \ \forall P_j \in E_k \).

The Production Rules with Exceptions and Fuzzy Hierarchy for each class \( D_k \ (1 \leq k \leq n) \) in the Freq matrix that has \( E_k \) as a non-empty set of exception condition(s) will have the following format:

\[
P \rightarrow D_k

\text{Unless } e_{k1} \lor ... \lor e_{ki} \lor ...

\text{Generality}[D_{k-1}]

\text{Specificity}[D_{k1}(d_1), ..., D_{kj}(d_j), ...] ....(5.8)
\]

where,

\[
P = (P_{pub})_k \cup (P_{spl})_k \cup (P_{pvt})_k,
\]

\( D_{k-1} \) is the general class of \( D_k \), and

\( e_{ki} \in E_k \) is the i-th exception condition to the class \( D_k \). The exception conditions in \( E_k \) are arranged in decreasing order of priority i.e. element \( \text{Freq}[e_{ki}, D_k] \) with highest
frequency value will be evaluated first (see the Maximum occurrences priority section 2.5.2 in chapter 2). Each exception condition \( e_{ki} \in E_k \) \((i = 1, 2, \ldots)\) is exception condition to each specific class \( D_{kj} \) \((j = 1, 2, \ldots)\). The Generality\([D_{k-1}]\) and Specificity\([D_{kl}(d_l), \ldots, D_{kj}(d_j), \ldots]\) relations have been discussed in chapter 4.

5.4. Discovery of Production Rules with Exceptions and Fuzzy Hierarchy \((D^{PREF\_H})\) Algorithm

\(D^{PREF\_H}\)-algorithm is an extension of \(D^{PR\_F\_H}\)-algorithm (chapter 4) and the steps which are identical to \(D^{PR\_F\_H}\)-algorithm are just marked "No change" and only the modified steps are described below:

Step 1.  
*No change.*

Step 2.  
*No change.*

Step 3.  
This step will be ignored because the exception conditions may be found in the irrelevant tuples i.e. small portion of data.

Step 4.  
*No change.*

Step 5.  
*No change.*

Step 6.  
For each weak class \( D_k \) \((1 \leq k \leq n)\) for which \((P_{unq})_k\) does not exist, the exception set \( E_k \) is discovered by the Exception Mining algorithm (given below). If the exception set \( E_k \) to the class \( D_k \) exists then all the exceptions in \( E_k \) are reordered using the maximum occurrences priority. Also,

\[
\text{If} \quad (P_{unq})_k = \emptyset \quad \text{then} \\
E_k = \text{Exceptions\_Mining}(D_k).
\]

Step 7.  
Reduce the size of the Freq matrix by removing all the classes having no highlighted elements corresponding to the discovered root properties. Further, reduce the size of the Freq matrix by removing each class \( D_k \) for which \((P_{unq})_k = E_k = \emptyset\).

Step 8.  
The \((P_{unq})_r\) for the root class \( D_r \) in the Fuzzy Hierarchical structure may or may not exist. In case \((P_{unq})_r\) does not exist then all the properties of the class \( D_r \) may be inherited by a particular specific class, which means \( \deg\_\text{root}(D_r) = 1 \), and therefore the degree of the root class will be in the interval \((0.5 \ldots 1]\).
Step 9. *No change.*

Step 10. During the construction of the fuzzy hierarchy from DM, if the candidate class $D_j$ (in the item $D_i \in D_j$) becomes part of the fuzzy hierarchy then the common exceptions between $D_j$ and general class $D_k$ at the higher levels (i.e. any class on the path starting from root class $D_r$ to the class $D_i$) are removed from $D_j$. So,

\[
\text{For each class } D_k \in (\text{Path})_{r \rightarrow i} \\
E_j = E_j - (E_j \cap E_k).
\]

Also during the construction of the fuzzy hierarchy, we have to verify that the exception condition of any general class $D_k$ is not associated with the preconditions of the $D_j$. Further, the preconditions of $D_k$ are disjoint with the set of exception conditions $E_j$ as follows:

\[
\text{For any class } D_k \in (\text{Path})_{r \rightarrow i} \\
\{ \text{Consider } \alpha_k = (P_{pub})_k \cup (P_{split})_k \cup (P_{priv})_k, \text{ and} \\
\text{Consider } \alpha_j = (P_{pub})_j \cup (P_{split})_j \cup (P_{priv})_j. \\
\text{If } (E_k \land \alpha_j \in D_j) \text{ or } (\alpha_k \land E_j \neq \emptyset) \text{ then} \\
\text{If } k \neq r \text{ then} \\
D_r \text{ is the general class of } D_j \text{ i.e. } D_r \subset D_j. \\
\text{Else} \\
\text{If } k = r \text{ then } D_j \text{ is irrelevant class}.\}
\]

Step 11. *No change.*

5.4.1. Exceptions Mining

For a given weak class $D_a$, arranging the properties in $(P_{pub})_a$ in ascending order of probability values helps in discovering the best combination of properties that covers maximum instances in the class $D_a$.

Initializing the set $\alpha_a$ by the root properties $P_{root} \subseteq (P_{pub})_a$ helps in discovering the minimum set of premises for the weak class $D_a$. If the exception set $E_a$ to the class $D_a$ does not exist i.e. $E_a = \emptyset$ with the current $\alpha_a$ then a single property $P_i \in (P_{pub})_a$ is added to $\alpha_a$ and removed from $(P_{pub})_a$. The process of increasing $\alpha_a$ and decreasing $(P_{pub})_a$ is
repeated until $E_a$ exists or $(P_{pub})_a$ becomes empty. If $E_a$ exists then the $(P_{unq})_a = \alpha_a$, and the highlighted elements corresponding to the properties which are not included in $(P_{unq})_a$ are converted to unhighlighted elements. Otherwise, if $(P_{pub})_a$ becomes empty and $E_a$ does not exist then class $D_a$ declared as an irrelevant class.

5.4.1.1. Exceptions Mining Algorithm

**Input**: Freq matrix,

Weak class $D_a$ (i.e. $(P_{pub})_a \neq \emptyset$ and $(P_{unq})_a = \emptyset$),

$P_{root}$ (non-empty set of root properties).

**Output**: $(P_{unq})_a$, $E_a$ (set of exception condition to the class $D_a$),

Freq matrix (Possible modified with some highlighted elements converted to unhighlighted elements).

**Method**:

(1) Initialization:

$$E_a \leftarrow \emptyset,$$

$$\alpha_a \leftarrow P_{root},$$

$$E^i_a \leftarrow \emptyset \ (i = 1 \ldots n) \ \forall \ i \neq a,$$

$$(P_{pub})_a \leftarrow (P_{pub})_a - \alpha_a,$$

exception\_found $\leftarrow$ True. //Boolean variable to ensure whether $E_a$ is discovered or no.

(2) Arrange the properties in $(P_{pub})_a$ in ascending order of probability values.

(3) For each class $D_b \neq D_a \ (b = 1 \ldots n)$

(3.1) If $| \alpha_a \land D_a | \gg | \alpha_a \land D_b |$ and $| \alpha_a \land D_b | > 0$ then

(3.1.1) For each property $P_j \notin (P_{pub})_a$ and $P_j \notin \alpha_a \ (j = 1 \ldots m)$

If $(\alpha_a \land P_j$ does not belong to class $D_a$ and $Freq[P_j, D_b] > 0$ and $\alpha_a \land P_j$ belongs to class $D_b)$

Then $E^b_a = E^b_a \cup \{P_j\}$ // exception condition discovered.

(3.1.2) If $E^b_a$ covers all the instances $\alpha_a \land D_b$ then

$$E_a = E_a \cup E^b_a$$ // set of exception conditions discovered from class $D_b$.

Else { exception\_found = False.

Go to step (4) // breaking the outer loop of step (3)}.

(4) If $(P_{pub})_a = \emptyset$ and exception\_found = False then
\{ E_a = \emptyset. \\
(P_{\text{unq}})_a = \emptyset \\
\text{Go to step (6) } \quad //\text{stop} \\
\} \text{Else if } (P_{\text{pub}})_a \neq \emptyset \text{ and } \text{exception\_found} = \text{False} \text{ then} \\
\{ E'_a = \emptyset \quad (i = 1...n). \\
\alpha_a = \alpha_a \cup \{ P_i \} , \ P_i \in (P_{\text{pub}})_a. \\
(P_{\text{pub}})_a = (P_{\text{pub}})_a - \{ P_i \}. \\
\text{exception\_found} = \text{True.} \\
\text{Go to step (3) } \quad //\text{repeat the process for the new set } \alpha_a} \\
(5) \text{If } E_a \neq \emptyset \text{ then} \\
(5.1) \text{For each property } P_i \in (P_{\text{pub}})_a. \\
\text{Convert the highlighted element } \text{Freq}[P_i, D_a] \text{ to unhighlighted element.} \\
(5.2) \ (P_{\text{unq}})_a = \alpha_a. \\
(6) \text{Return } (P_{\text{unq}})_a, \ Freq \text{ matrix and } E_a.

Applying \text{DPREFH-} \text{algorithm on the Freq matrix given in Fig.4.14 (chapter 4), the discovered set of exception conditions to the weak classes } D_{14} \text{ and } D_{15} \text{ are } E_{14} = \{ P_{25} \} \text{ and } E_{15} = \{ P_{16} \} \text{ respectively. The Degree Matrix (DM) is given below (Fig.5.3):}

\[
\begin{array}{cccccccccccc}
 & D_1 & D_2 & D_3 & D_4 & D_5 & D_7 & D_8 & D_9 & D_{10} & D_{11} & D_{12} & D_{13} & D_{14} & D_{15} \\
D_1 & 0 & 0.67 & 0.5 & 0.5 & 0.5 & 0.67 & 0.5 & 0.83 & 0.5 & 0.5 & 0.5 & 0.5 & 0.6 & 0.66 \\
D_2 & 0.67 & 0 & 0.5 & 0.67 & 0.83 & 0.5 & 0.67 & 0.67 & 0.5 & 0.5 & 0.67 & 0.67 & 0.67 & 0.66 \\
D_3 & 0.5 & 0.5 & 0 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.66 \\
D_4 & 0 & 0 & 0 & 0.71 & 0.57 & 0 & 0.71 & 0.57 & 0 & 0.71 & 0.71 & 0 & 0.66 & 0.66 \\
D_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.88 & 0.88 & 0 & 0.88 & 0.88 & 0 & 0 \\
D_6 & 0.8 & 0.8 & 0.6 & 0.6 & 0.6 & 0.8 & 0.8 & 0.8 & 0.6 & 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\
D_7 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\
D_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0.57 & 0 & 0.57 & 0 & 0.57 & 0 & 0 & 0.66 \\
D_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.66 \\
D_{10} & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0.57 & 0 & 0 & 0 & 0 & 0.66 \\
D_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.66 \\
D_{12} & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.66 \\
D_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.57 & 0 & 0 & 0 & 0 & 0.66 \\
D_{14} & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0.66 \\
D_{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.66 \\
\end{array}
\]

\text{Fig.5.3: Degree Matrix (DM) with weak classes for the synthetic data.}

The Fuzzy Hierarchical structure with Exceptions is given below (Fig.5.4):
Applying the `DPREFHL`-algorithm on the Freq matrix (Fig.4.8) of the zoo dataset (chapter 4), the discovered set of exception conditions to the weak class G is $E_G = \{\text{Backbone}\}$ and the unique properties are $(P_{\text{uniq}})_G = \{\text{Eggs, Aquatic, Predator}\}$. If we consider only the subsumption degree for constructing the fuzzy hierarchy then class E becomes specific class of G as shown in Fig.5.5.

Eggs_Animal [Eggs]

Referring to both Fig.5.5 and the Freq matrix given in Fig.4.8 (chapter 4), the property "Backbone" is an exception condition to the class G. However, $\text{Freq}[\text{Backbone}, E] = 1$, the property "Backbone" is associated with the properties "Leg = 4" and "Toothed" of the class E. And hence, class E cannot become specific class of G. Applying the code at step 10 of the `DPREFHL`-algorithm will produce the final Fuzzy Hierarchical structure with Exception as shown in Fig.5.6. 
Example 5.1

This example illustrates the discovery of multiple exception conditions from the Freq matrix-I given in Fig.5.7. Assume that the threshold value = 0.6, |D_a| = 100 and |D_b| = 5.

Table 5.1 gives five instances of data corresponding to the class D_b. All the properties P_1, P_2, ..., P_5 in the Freq matrix (Fig.5.7) are assumed to be attributes of Boolean type and symbol ‘✓’ indicates true value of the properties.

<table>
<thead>
<tr>
<th>Instance</th>
<th>P_1</th>
<th>P_2</th>
<th>P_3</th>
<th>P_4</th>
<th>P_5</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance 1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>D_b</td>
</tr>
<tr>
<td>Instance 2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>D_b</td>
</tr>
<tr>
<td>Instance 3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>D_b</td>
</tr>
<tr>
<td>Instance 4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>D_b</td>
</tr>
<tr>
<td>Instance 5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>D_b</td>
</tr>
</tbody>
</table>

Referring to the Freq matrix-I in Fig.5.7,

(P_pub)_a = \{P_1, P_3\}.

(P_unq)_b = \{P_1, P_4\} /* Assuming P_1 \land P_4 does not belong to class D_a */.

P_root = \{P_1\}.

Initializing \(\alpha_a = P_{root} = \{P_1\}\) and decreasing \((P_{pub})_a = \{P_3\}\). The condition in the step (3.1) is satisfied i.e.

\[ |P_1 \land D_a| \gg |P_1 \land D_b| \text{ and } |P_1 \land D_b| > 0 \text{ i.e. } 0.9 \times 100 \gg 0.8 \times 4.\]
Considering $P_j = P_2$ and verifying the conditions in the step (3.1.1) as follows:

(i) $\text{Freq}[P_2, D_a] = 0$.

(ii) $\text{Freq}[P_2, D_b] > 0$.

(iii) $P_2 \land \alpha_a$ (i.e. $P_2 \land P_1$) belongs to class $D_b$ (instance 1 in Table 5.1).

From (i), (ii) and (iii) the exception set $E_a^b = \{P_2\}$. Further, no other exception can be discovered from the class $D_b$.

The condition in the step (3.1.2) is not satisfied i.e. all the instances having $\alpha_a = \{P_1\}$ in the class $D_b$ are not covered by the exception condition $P_2$ in the set $E_a^b$ (instances 2, 3 and 5 in Table 5.1). Therefore, a property $P_3 \in (P_{\text{pub}})_a$ is moved to the set $\alpha_a$ and the process is repeated for the new $\alpha_a = \{P_1, P_3\}$. There exist two instances in class $D_b$ (1 and 3, in Table 5.1) having the combination of the properties in $\alpha_a = \{P_1, P_3\}$. These two instances are covered by the exception condition $E_a^b = \{P_2, P_3\}$ and hence the discovered set of premises for the class $D_a$ is $(P_{\text{unq}})_a = \alpha_a = \{P_1, P_3\}$. The weak class $D_a$ becomes strongly relevant class and it becomes part of the fuzzy hierarchy as given in Fig.5.8.

![Fig.5.8: Fuzzy hierarchy structure with multiple exception conditions.]

**Example 5.2**

This example illustrates how some highlighted elements in the Freq matrix-II (Fig.5.9) are converted to unhighlighted elements. Assume that threshold value = 0.6, $|D_a| = 100$ and $|D_b| = 4$.

![Fig.5.9: Freq matrix-II for a root class with exception condition]
Table 5.2 gives four instances of data corresponding to the class $D_b$. All the properties $P_1$, $P_2$, ..., $P_6$ in the Freq matrix-II (Fig.5.9) are assumed to be attributes of Boolean type and symbol ‘\' indicates true value of the properties.

<table>
<thead>
<tr>
<th>Instance</th>
<th>$P_1$</th>
<th>$P_2$</th>
<th>$P_3$</th>
<th>$P_4$</th>
<th>$P_5$</th>
<th>$P_6$</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance 1</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td>$D_b$</td>
</tr>
<tr>
<td>Instance 2</td>
<td></td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td>$D_b$</td>
</tr>
<tr>
<td>Instance 3</td>
<td>√</td>
<td></td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td>$D_b$</td>
</tr>
<tr>
<td>Instance 4</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$D_b$</td>
</tr>
</tbody>
</table>

Referring to the Freq matrix-II in Fig.5.9,

$$(P_{\text{pub}})_a = \{P_1, P_3, P_6\}.$$  

$$(P_{\text{uniq}})_b = \{P_1, P_3, P_4\} \quad /*$ Assuming $P_1 \land P_3 \land P_4$ does not belong to class $D_a */.$

$P_{\text{root}} = \{P_1, P_3\}.$

Initializing $\alpha_a = P_{\text{root}} = \{P_1, P_3\}$. All the instances having $\alpha_a = \{P_1, P_3\}$ in the class $D_b$ are covered by the discovered set of exception conditions $E^b_a = \{P_2, P_5\}$ (instances 1 and 3 in Table 5.2) and hence the discovered premises for the class $D_a$ is $\alpha_a = \{P_1, P_3\}$.

The remaining property $\{P_6\}$ in $(P_{\text{pub}})_a$ is removed to get $(P_{\text{pub}})_a = \{P_1, P_3\}$ and the highlighted element $\text{Freq}[P_6, D_a]$ in the Freq matrix (Fig.5.9) is converted to unhighlighted element (Fig.5.10).

<table>
<thead>
<tr>
<th>$D_a$</th>
<th>$D_b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>0.8</td>
</tr>
<tr>
<td>$P_2$</td>
<td>0.0</td>
</tr>
<tr>
<td>$P_3$</td>
<td>0.8</td>
</tr>
<tr>
<td>$P_4$</td>
<td>0.1</td>
</tr>
<tr>
<td>$P_5$</td>
<td>0.0</td>
</tr>
<tr>
<td>$P_6$</td>
<td>0.6</td>
</tr>
</tbody>
</table>

Fig.5.10: Freq matrix-III, converting highlighted element to unhighlighted.

Since both the classes $D_a$ and $D_b$ logically contradict each other and $D_a$ is the root class, class $D_b$ cannot be a specific class of $D_a$. Therefore, the hierarchy will contain only the root node as shown in Fig.5.11.

$$D_a [P_1, P_3] \n (P_2 \lor P_3)$$

Fig.5.11: Hierarchy with root class only.
Example 5.3

This example illustrates the discovery of exception conditions from different classes given in the Freq matrix-V Fig.5.12. Assume that threshold value = 0.6, \(|D_a| = 100, |D_b| = |D_c| = 3.\)

\[
\begin{array}{ccc}
  P_1 & D_a & \mathbf{1.0} & D_b & \mathbf{0.3} & D_c & \mathbf{0.66} \\
  P_2 & 0.0 & \mathbf{1.0} & 0.33 \\
  P_3 & 0.0 & \mathbf{0.66} & 0.33 \\
  P_4 & 0.2 & 0.33 & \mathbf{0.66} \\
\end{array}
\]

Fig.5.12: Freq matrix-V

Table 5.3 gives three instances of data corresponding to the class \(D_b\) and Table 5.4 gives three instances of data corresponding to the class \(D_c\). All the properties \(P_1, P_2, \ldots, P_4\) in the Freq matrix-V (Fig.5.12) are assumed to be attributes of Boolean type and symbol ‘\(\checkmark\)’ indicates true value of the properties.

<table>
<thead>
<tr>
<th>Instance</th>
<th>(P_1)</th>
<th>(P_2)</th>
<th>(P_3)</th>
<th>(P_4)</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance 1</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td></td>
<td></td>
<td>(D_b)</td>
</tr>
<tr>
<td>Instance 2</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td></td>
<td></td>
<td>(D_b)</td>
</tr>
<tr>
<td>Instance 3</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td></td>
<td>(D_b)</td>
</tr>
</tbody>
</table>

Table 5.4: Data corresponding to class \(D_c\) in Freq matrix-V.

<table>
<thead>
<tr>
<th>Instance</th>
<th>(P_1)</th>
<th>(P_2)</th>
<th>(P_3)</th>
<th>(P_4)</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance 1</td>
<td>(\checkmark)</td>
<td></td>
<td></td>
<td></td>
<td>(D_c)</td>
</tr>
<tr>
<td>Instance 2</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td></td>
<td>(D_c)</td>
</tr>
<tr>
<td>Instance 3</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td></td>
<td></td>
<td>(D_c)</td>
</tr>
</tbody>
</table>

Referring to the Freq matrix-V in Fig.5.12,

\((P_{pub})_a = \{P_1\}\).

\((P_{unq})_c = \{P_1, P_3\}\).

\(P\text{\tiny root} = \{P_1\}\).

Initializing \(\alpha_a = P\text{\tiny root} = \{P_1\}\). All the instances having \(\alpha_a = \{P_1\}\) in the class \(D_b\) are covered by the discovered set of exception conditions \(E^b_a = \{P_2\}\) (instance 1 in Table 5.3) and all the instances having \(\alpha_a = \{P_1\}\) in the class \(D_c\) are covered by the discovered set of exception conditions \(E^c_a = \{P_3\}\) (instances 1 and 2 in Table 5.4). The exception conditions to the class \(D_a\) is \(E_a = E^c_a \cup E^b_b = \{P_2, P_3\}\), and the discovered set of premises for the class \(D_a\) is \((P_{unq})_a = \alpha_a = \{P_1\}\).
Since the exception condition \( P_3 \) to class \( D_a \) is a member of the premise set \( (P_{\text{unq}})_c = \{P_1, P_3\} \) of the class \( D_c \), class \( D_c \) cannot become a specific class of class \( D_a \) even though both the classes have a common property \( P_1 \). Therefore, two hierarchies are constructed each one with single node (root node) as given in Fig.5.13 ((a) and (b)).

\[
D_a [P_1] \sqcup (P_2 \lor P_3) \quad \quad \quad D_c [P_1, P_3]
\]

(a) Root class with multiple exceptions  (b) Root class without exceptions discovered from different classes.

Fig.5.13: Construction of two hierarchies each with single node.

**Example 5.4**

This example illustrates the construction of Fuzzy Hierarchy structure from a Freq matrix having all weak classes. Table 5.5 contains different classes of real-life data given in the HCPR-tree [11, 70]. The Freq matrix for this data is given in Fig.5.14 with the assumption that each class is of length 100 and the given *threshold* value is 0.6.

<table>
<thead>
<tr>
<th>Class name</th>
<th>Symbol in Freq matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lives_in_city (x,y)</td>
<td>( D_1 )</td>
</tr>
<tr>
<td>At home (x)</td>
<td>( D_2 )</td>
</tr>
<tr>
<td>Outside home (x)</td>
<td>( D_3 )</td>
</tr>
<tr>
<td>Working outdoor (x)</td>
<td>( D_4 )</td>
</tr>
<tr>
<td>Entertaining outdoor (x)</td>
<td>( D_5 )</td>
</tr>
<tr>
<td>Is outside city (x,y)</td>
<td>( D_6 )</td>
</tr>
<tr>
<td></td>
<td>( D_1 ) ( D_2 ) ( D_3 ) ( D_4 ) ( D_5 ) ( D_6 )</td>
</tr>
<tr>
<td>Lives_in_city</td>
<td>( P_1 )</td>
</tr>
<tr>
<td>Time(night)</td>
<td>( P_2 )</td>
</tr>
<tr>
<td>Time(day)</td>
<td>( P_3 )</td>
</tr>
<tr>
<td>Day(work)</td>
<td>( P_4 )</td>
</tr>
<tr>
<td>Day(Sunday)</td>
<td>( P_5 )</td>
</tr>
<tr>
<td>Is_on_tour</td>
<td>( P_6 )</td>
</tr>
<tr>
<td>Doing_overtime</td>
<td>( P_7 )</td>
</tr>
<tr>
<td>Work_in_night_shift</td>
<td>( P_8 )</td>
</tr>
<tr>
<td>Is_ill</td>
<td>( P_9 )</td>
</tr>
<tr>
<td>Bad_weather</td>
<td>( P_{10} )</td>
</tr>
<tr>
<td>Holiday</td>
<td>( P_{11} )</td>
</tr>
<tr>
<td>Unemployed</td>
<td>( P_{12} )</td>
</tr>
<tr>
<td>Met_an_accident</td>
<td>( P_{13} )</td>
</tr>
</tbody>
</table>

Fig.5.14: Freq matrix for real-life data (*threshold* = 0.6)
Referring to the Freq matrix in Fig.5.14, the following information can be extracted regarding combination of some properties which does not belong to specific class(es):

(i) \( P_2 \land P_i \notin D_1 \) \((i=7, 8)\),

(ii) \( P_3 \land P_5 \notin D_j \) \((j=1, 2)\), and

(iii) \( P_4 \land P_i \notin D_j \) \((i = 2, 3, 11, 12. j = 2, 3, 4)\).

The root property is \( P_{\text{root}} = \{P_1\} \) and the discovered root class is \( D_1[P_1] \). The element \( \text{Freq}[P_1, D_6] \) is unhighlighted element, therefore the class \( D_6 \) is an irrelevant class. The remaining classes are weak classes because \( (P_{\text{unq}})_i = \emptyset \) \((i=1...5)\). The degree matrix \((\text{DM})\) is given below (Fig.5.15):

\[
\begin{array}{cccc}
\text{D} & \text{D}_2 & \text{D}_3 & \text{D}_4 & \text{D}_5 \\
\text{D}_1 & 1 & 1 & 1 & 1 \\
\text{D}_2 & 0 & 0 & 0 & 0 \\
\text{D}_3 & 0 & 0 & 1 & 1 \\
\text{D}_4 & 0 & 0 & 0 & 0.66 \\
\text{D}_5 & 0 & 0 & 0.66 & 0 \\
\end{array}
\]

Fig.5.15: DM for real-life data.

Class \( D_2 \) is discovered to be a leaf class of the fuzzy hierarchy. The above degree matrix \((\text{DM})\) can be reduced by removing class \( D_2 \). The final fuzzy hierarchical structure with Exceptions is given in Fig.5.16.

\[
\begin{array}{l}
\text{x is in city}\ y [\text{Lives in city}] \land (\text{is on tour}) \\
\quad 1 \\
\quad 1 \\
\text{x outside home} [\text{Time(day)}] \land (\text{Is ill} \lor \text{Bad weather}) \\
\qquad 1 \\
\text{x working outdoor} [\text{Day(work)}] \land (\text{Holiday} \lor \text{Unemployed}) \\
\qquad 1 \\
\text{x at home} [\text{Time(night)}] \land (\text{Doing overtime} \lor \text{Working in night shift}) \\
\qquad 1 \\
\text{x entertaining outdoor} [\text{Day(Sunday)}] \land (\text{Met an accident}) \\
\end{array}
\]

Fig.5.16: Fuzzy Hierarchical structure with exceptions for real-life data.

The discovered Production Rules with Exceptions and the Fuzzy Hierarchy (Fig.5.16) are:

(1) \( \text{Lives in city}(x) \rightarrow \text{is in city}(x, y) \)
Unless [is_on_tour]
Generality[ ]
Specificity[Outside_home(x) (1), At_home(x) (1)].

(2) Time(night) → At_home(x)
    Unless [Doing_overtime ∨ Working_in_night_shift]
    Generality[Lives_in_city]
    Specificity[ ].

(3) Time(day) → Outside_home(x)
    Unless [Is_ill ∨ Bad_weather]
    Generality[Lives_in_city]
    Specificity[Working_outdoor(x) (1), Entreating_outdoor(x) (1)].

(4) Day(work) → Working_outdoor(x)
    Unless [Holiday ∨ Unemployed]
    Generality[Outside_home(x)]
    Specificity[ ].

(5) Day(Sunday) → Entertaining_outdoor(x)
    Unless [Met_an_accident]
    Generality[Outside_home(x)]
    Specificity[ ].

100