Introduction to Rough Sets

Davide Ciucci

Dipartimento di Informatica, Sistemistica e Comunicazione
Università di Milano Bicocca

PhD 2021/22

Outline

Outline

What is a Rough *set?*

Simple question, difficult answer...

What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, ...
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, ...
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, ...
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, ...
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, ...
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, . . .
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, . . .
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

What is a Rough *set?*

Simple question, difficult answer...
What is a Fuzzy Subset of X? $f : X \mapsto [0, 1]$

- We need several notions: indiscernibility, granulation of the universe, approximations, ...
- More than one definition is possible... some "ingredients" are
  - A set H whose elements are known (extension), but we are not able to describe it (intension)
  - We are able to give (intension and extension) a pair of sets which are an approximation of H

Rough set theory includes some tools for knowledge discovery: reducts (feature selection) and rules

Outline

Information Table - example

HA = Head Ache
MP = Muscle Pain

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

Information Table - example

HA = Head Ache
MP = Muscle Pain

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

Information Table - example

HA = Head Ache
MP = Muscle Pain

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

Information Table - definition

Definition (Information Table or Information System)

$\mathcal{S}(U) = \langle U, Att, Val, F \rangle$

U set of objects

Att set of attributes

Val set of possible values for the attributes

$F : U \times A \mapsto V$ function that assigns to each object a value for any attribute

Sometimes: Val_a with $a \in Att$

In the example: objects ={P1, ..., P5}, Attributes = {Pressure, HA, Temperature, MP}, Val ={Yes, No, 37-38, ...}

$F(P2, Pressure) = High$

Information Table - definition

Definition (Information Table or Information System)

$\mathcal{S}(U) = \langle U, Att, Val, F \rangle$
U set of objects
Att set of attributes
Val set of possible values for the attributes
$F : U \times A \mapsto V$ function that assigns to each object a value for any attribute

Sometimes: Val_a with $a \in Att$

In the example: objects ={P1, ..., P5}, Attributes = {Pressure, HA, Temperature, MP}, Val ={Yes, No, 37-38, ...}
$F(P2, Pressure) = High$

Information Table - definition

Definition (Information Table or Information System)

$\mathcal{S}(U) = \langle U, Att, Val, F \rangle$
U set of objects
Att set of attributes
Val set of possible values for the attributes
$F : U \times A \mapsto V$ function that assigns to each object a value for any attribute

Sometimes: Val_a with $a \in Att$

In the example: objects ={P1, . . . , P5}, Attributes = {Pressure, HA, Temperature, MP}, Val ={Yes, No, 37-38, . . .}
$F(P2, Pressure) = High$

Information Table - definition

Definition (Information Table or Information System)

$\mathcal{S}(U) = \langle U, Att, Val, F \rangle$
U set of objects
Att set of attributes
Val set of possible values for the attributes
$F : U \times A \mapsto V$ function that assigns to each object a value for any attribute

Sometimes: Val_a with $a \in Att$

In the example: objects ={P1, . . . , P5}, Attributes = {Pressure, HA, Temperature, MP}, Val ={Yes, No, 37-38, . . . }
$F(P2, Pressure) = High$

Information Table - definition

Definition (Information Table or Information System)

$\mathcal{S}(U) = \langle U, Att, Val, F \rangle$
U set of objects
Att set of attributes
Val set of possible values for the attributes
$F : U \times A \mapsto V$ function that assigns to each object a value for any attribute

Sometimes: Val_a with $a \in Att$

In the example: objects ={P1, . . . , P5}, Attributes = {Pressure, HA, Temperature, MP}, Val ={Yes, No, 37-38, . . .}
$F(P2, Pressure) = High$

Information Table - definition

Definition (Information Table or Information System)

$\mathcal{S}(U) = \langle U, Att, Val, F\rangle$
U set of objects
Att set of attributes
Val set of possible values for the attributes
$F : U \times A \mapsto V$ function that assigns to each object a value for any attribute

Sometimes: Val_a with $a \in Att$

In the example: objects ={P1, . . . , P5}, Attributes = {Pressure, HA, Temperature, MP}, Val ={Yes, No, 37-38, . . .}
$F(P2, Pressure) = High$

Decision System - Example

HA = Head Ache
MP = Muscle Pain

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

Decision Classes: $U_A = \{P1\}$, $U_B = \{P3\}$, $U_{NO} = \{P2, P4, P5\}$

P2, P3: same symptoms, different disease $\rightarrow$ the system is inconsistent

Decision System - Example

HA = Head Ache
MP = Muscle Pain

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

Decision Classes: $U_A = \{P1\}$, $U_B = \{P3\}$, $U_{NO} = \{P2, P4, P5\}$

P2, P3: same symptoms, different disease $\rightarrow$ the system is inconsistent

Decision System - Example

HA = Head Ache
MP = Muscle Pain

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

Decision Classes: $U_A = \{P1\}$, $U_B = \{P3\}$, $U_{NO} = \{P2, P4, P5\}$

P2, P3: same symptoms, different disease $\rightarrow$ the system is inconsistent

Definitions

Definition (Decision System)

$\mathcal{S}(U) = \langle U, C \cup \{d\}, Val, F \rangle$
U set of objects
C set of condition attributes
d decision attribute
Val set of possible values for the attributes
$F : U \times C \cup \{d\} \mapsto V$ function that assigns to each object a value for any attribute

Definition (Consistent Decision System)

There are no two objects $O_1, O_2 \in U$ with same value for condition attributes and different decision

Definitions

Definition (Decision System)

$\mathcal{S}(U) = \langle U, C \cup \{d\}, Val, F \rangle$
U set of objects
C set of condition attributes
d decision attribute
Val set of possible values for the attributes
$F : U \times C \cup \{d\} \mapsto V$ function that assigns to each object a value for any attribute

Definition (Consistent Decision System)

There are no two objects $O_1, O_2 \in U$ with same value for condition attributes and different decision

Definitions

Definition (Decision System)

$\mathcal{S}(U) = \langle U, C \cup \{d\}, Val, F\rangle$
U set of objects
C set of condition attributes
d decision attribute
Val set of possible values for the attributes
$F : U \times C \cup \{d\} \mapsto V$ function that assigns to each object a value for any attribute

Definition (Consistent Decision System)

There are no two objects $O_1, O_2 \in U$ with same value for condition attributes and different decision

Outline

Indiscernibility relation - example

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

P2 and P3 same values for all attributes: they are indiscernible (indistinguishable, equivalent, ...)

{P2, P3} is a granule of information

A partition of the universe: Π = {P1}, {P2,P3}, {P4}, {P5}

Indiscernibility relation - example

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

P2 and P3 same values for all attributes: they are indiscernible (indistinguishable, equivalent, ...)

{P2, P3} is a granule of information

A partition of the universe: Π = {P1}, {P2,P3}, {P4}, {P5}

Indiscernibility relation - example

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

P2 and P3 same values for all attributes: they are indiscernible (indistinguishable, equivalent, ...)

{P2, P3} is a granule of information

A partition of the universe: Π = {P1}, {P2,P3}, {P4}, {P5}

Indiscernibility relation - definition

Definition (Indiscernibility)

Given a set of attributes $A \subseteq Att$
two objects $x, y \in U$ are indiscernible with respect to A if

$$\forall a \in A \quad F(a, x) = F(a, y)$$

In this case we write xI_Ay

I_A is an equivalence relation: reflexive, symmetric, transitive
I_A partitions U in equivalence classes (the granules of information)

$$[x]_A := \{y \in U : xI_Ay\}$$

Indiscernibility relation - definition

Definition (Indiscernibility)

Given a set of attributes $A \subseteq Att$
two objects $x, y \in U$ are indiscernible with respect to A if

$$\forall a \in A \quad F(a, x) = F(a, y)$$

In this case we write $xI_A y$

I_A is an equivalence relation: reflexive, symmetric, transitive
I_A partitions U in equivalence classes (the granules of information)

$$[x]_A := \{y \in U : xI_A y\}$$

Indiscernibility relation - definition

Definition (Indiscernibility)

Given a set of attributes $A \subseteq Att$
two objects $x, y \in U$ are indiscernible with respect to A if

$$\forall a \in A \quad F(a,x) = F(a,y)$$

In this case we write xI_Ay

I_A is an equivalence relation: reflexive, symmetric, transitive
I_A partitions U in equivalence classes (the granules of information)

$$[x]_A := \{y \in U : xI_Ay\}$$

Indiscernibility relation - definition

Definition (Indiscernibility)

Given a set of attributes $A \subseteq Att$
two objects $x, y \in U$ are indiscernible with respect to A if

$$\forall a \in A \quad F(a,x) = F(a,y)$$

In this case we write xI_Ay

I_A is an equivalence relation: reflexive, symmetric, transitive
I_A partitions U in equivalence classes (the granules of information)

$$[x]_A := \{y \in U : xI_Ay\}$$

Approximations - example

Partition {P1}, {P2,P3}, {P4}, {P5}

- The set $H = \{P1, P2, P3\}$ is the union of two equivalence classes $\{P1\} \cup \{P2, P3\}$
- The set $K = \{P1, P2\}$ is not
- H is exact, K is rough
- K can be approximated by a pair of exact sets: {P1}, {P1,P2,P3}

$$\{P1\} \subseteq K \subseteq \{P1, P2, P3\}$$

Approximations - example

Partition {P1}, {P2,P3}, {P4}, {P5}

- The set $H = \{P1, P2, P3\}$ is the union of two equivalence classes $\{P1\} \cup \{P2, P3\}$
- The set $K = \{P1, P2\}$ is not
- H is exact, K is rough
- K can be approximated by a pair of exact sets: {P1}, {P1,P2,P3}

$$\{P1\} \subseteq K \subseteq \{P1, P2, P3\}$$

Approximations - example

Partition {P1}, {P2,P3}, {P4}, {P5}

- The set $H = \{P1, P2, P3\}$ is the union of two equivalence classes $\{P1\} \cup \{P2, P3\}$
- The set $K = \{P1, P2\}$ is not
- H is exact, K is rough
- K can be approximated by a pair of exact sets: {P1}, {P1,P2,P3}

$$\{P1\} \subseteq K \subseteq \{P1, P2, P3\}$$

Approximations - example

Partition {P1}, {P2,P3}, {P4}, {P5}

- The set $H = \{P1, P2, P3\}$ is the union of two equivalence classes $\{P1\} \cup \{P2, P3\}$
- The set $K = \{P1, P2\}$ is not
- H is exact, K is rough
- K can be approximated by a pair of exact sets: {P1}, {P1,P2,P3}

$$\{P1\} \subseteq K \subseteq \{P1, P2, P3\}$$

Approximations - definition

Definition (Approximations)

Let $S(U) = \langle U, Att, val(U), F\rangle$ be an information table (a decision system)
Given a set of attributes $A \subseteq Att$, then for any set of objects $H \subseteq U$ we define
the lower approximation of H:

$$L(H) := \{x : [x]_A \subseteq H\}$$

the upper approximation of H:

$$U(H) := \{x : [x]_A \cap H \neq \emptyset\}$$

The pair $r(H) = \langle L(H), U(H)\rangle$ is named *rough approximation* (or rough set).

Approximations - definition

Definition (Approximations)

Let $S(U) = \langle U, Att, val(U), F\rangle$ be an information table (a decision system)
Given a set of attributes $A \subseteq Att$, then for any set of objects $H \subseteq U$ we define
the lower approximation of H:

$$L(H) := \{x : [x]_A \subseteq H\}$$

the upper approximation of H:

$$U(H) := \{x : [x]_A \cap H \neq \emptyset\}$$

The pair $r(H) = \langle L(H), U(H)\rangle$ is named *rough approximation* (or rough set)

Approximations - definition

Definition (Approximations)

Let $S(U) = \langle U, Att, val(U), F\rangle$ be an information table (a decision system)
Given a set of attributes $A \subseteq Att$, then for any set of objects $H \subseteq U$ we define
the lower approximation of H:

$$L(H) := \{x : [x]_A \subseteq H\}$$

the upper approximation of H:

$$U(H) := \{x : [x]_A \cap H \neq \emptyset\}$$

The pair $r(H) = \langle L(H), U(H)\rangle$ is named *rough approximation* (or rough set)

Approximations - definition

Definition (Approximations)

Let $S(U) = \langle U, Att, val(U), F \rangle$ be an information table (a decision system)
Given a set of attributes $A \subseteq Att$, then for any set of objects $H \subseteq U$ we define
the lower approximation of H:

$$L(H) := \{x : [x]_A \subseteq H\}$$

the upper approximation of H:

$$U(H) := \{x : [x]_A \cap H \neq \emptyset\}$$

The pair $r(H) = \langle L(H), U(H) \rangle$ is named *rough approximation* (or rough set)

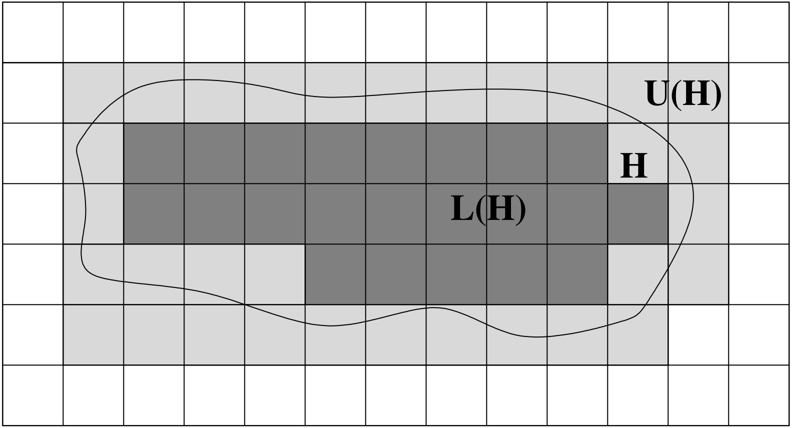

Figura: Lower and Upper approximations. Each square represents an equivalence class

Further regions

Exterior $E(H) = U^c(H)$ $L(H) \cap E(H) = \emptyset$
Rough approximation: $(L(H), E(H))$

Boundary $Bnd(H) = U(H) \setminus L(H)$

Interpretation	
Lower	sure belong to H
Exterior	sure not belong to H
Boundary	uncertain

Further regions

Exterior $E(H) = U^c(H)$ $L(H) \cap E(H) = \emptyset$
Rough approximation: $(L(H), E(H))$

Boundary $Bnd(H) = U(H) \setminus L(H)$

Interpretation	
Lower	sure belong to H
Exterior	sure not belong to H
Boundary	uncertain

Further regions

Exterior $E(H) = U^c(H)$ $L(H) \cap E(H) = \emptyset$
Rough approximation: $(L(H), E(H))$

Boundary $Bnd(H) = U(H) \setminus L(H)$

Interpretation	
Lower	sure belong to H
Exterior	sure not belong to H
Boundary	uncertain

Measures of Uncertainty

Accuracy

$$\alpha(H) = \frac{|L(H)|}{|U(H)|}$$

Roughness

$$1 - \alpha(H) = \frac{|Bnd(H)|}{|U(H)|}$$

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Approximation properties

- $L(\emptyset) = \emptyset \quad L(U) = U$
- $L(H) \subseteq H \quad H \subseteq U(H)$
- $L(H \cap K) = L(H) \cap L(K) \quad L(H) \cup L(K) \subseteq L(H \cup K)$
- $H \subseteq K$ implies $L(H) \subseteq L(K)$
- $L(L(H)) = L(H) \quad L(U(H)) = U(H)$
- $L(H) = (U(H^c))^c$

- Topological properties: Lower as interior, upper as closure
- Modal properties (S_5): Lower as necessity, upper as possibility

Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A\, y \text{ and } F(y, d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A\, y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Generalized Decision - example

Patient	Pressure	HA	Temperature	MP	Disease	δ_{Att}
P1	Normal	yes	38–39	yes	A	A,
P2	High	no	36–37	yes	NO	B,NO
P3	High	no	36-37	yes	B	B,NO
P4	Low	yes	35–36	no	NO	NO
P5	Normal	yes	36–37	yes	NO	NO

Outline

Generic relation

- R a binary relation on U: $R \subseteq U \times U$
- Granule of information $g_R(x) = \{y \in U : x\, R\, y\}$
- Approximations

$$l_R(H) = \{x \in U : gr(x) \subseteq H\}$$
$$u_R(H) = \{x \in U : gr(x) \cap H \neq \emptyset\}$$

Properties

- $l_R(U) = U$, $u_R(\emptyset) = \emptyset$, l, u are monotone
- If R is serial: $l_R(H) \subseteq u_R(H)$, $u_R(U) = U$, $l_R(\emptyset) = \emptyset$
- If R is reflexive: $l_R(H) \subseteq H \subseteq u_R(H)$

Generic relation

- R a binary relation on U: $R \subseteq U \times U$
- Granule of information $g_R(x) = \{y \in U : x\,R\,y\}$
- Approximations

$$l_R(H) = \{x \in U : gr(x) \subseteq H\}$$
$$u_R(H) = \{x \in U : gr(x) \cap H \neq \emptyset\}$$

Properties

- $l_R(U) = U$, $u_R(\emptyset) = \emptyset$, l, u are monotone
- If R is serial: $l_R(H) \subseteq u_R(H)$, $u_R(U) = U$, $l_R(\emptyset) = \emptyset$
- If R is reflexive: $l_R(H) \subseteq H \subseteq u_R(H)$

Generic relation

- R a binary relation on U: $R \subseteq U \times U$
- Granule of information $g_R(x) = \{y \in U : x\,R\,y\}$
- Approximations

$$l_R(H) = \{x \in U : gr(x) \subseteq H\}$$
$$u_R(H) = \{x \in U : gr(x) \cap H \neq \emptyset\}$$

Properties

- $l_R(U) = U$, $u_R(\emptyset) = \emptyset$, l, u are monotone
- If R is serial: $l_R(H) \subseteq u_R(H)$, $u_R(U) = U$, $l_R(\emptyset) = \emptyset$
- If R is reflexive: $l_R(H) \subseteq H \subseteq u_R(H)$

Generic relation

- R a binary relation on U: $R \subseteq U \times U$
- Granule of information $g_R(x) = \{y \in U : x\, R\, y\}$
- Approximations

$$l_R(H) = \{x \in U : gr(x) \subseteq H\}$$
$$u_R(H) = \{x \in U : gr(x) \cap H \neq \emptyset\}$$

Properties

- $l_R(U) = U$, $u_R(\emptyset) = \emptyset$, l, u are monotone
- If R is serial: $l_R(H) \subseteq u_R(H)$, $u_R(U) = U$, $l_R(\emptyset) = \emptyset$
- If R is reflexive: $l_R(H) \subseteq H \subseteq u_R(H)$

Generic relation

- R a binary relation on U: $R \subseteq U \times U$
- Granule of information $g_R(x) = \{y \in U : x\,R\,y\}$
- Approximations

$$l_R(H) = \{x \in U : gr(x) \subseteq H\}$$
$$u_R(H) = \{x \in U : gr(x) \cap H \neq \emptyset\}$$

Properties

- $l_R(U) = U$, $u_R(\emptyset) = \emptyset$, l, u are monotone
- If R is serial: $l_R(H) \subseteq u_R(H)$, $u_R(U) = U$, $l_R(\emptyset) = \emptyset$
- If R is reflexive: $l_R(H) \subseteq H \subseteq u_R(H)$

Generic relation

- R a binary relation on U: $R \subseteq U \times U$
- Granule of information $g_R(x) = \{y \in U : x\,R\,y\}$
- Approximations

$$l_R(H) = \{x \in U : gr(x) \subseteq H\}$$
$$u_R(H) = \{x \in U : gr(x) \cap H \neq \emptyset\}$$

Properties

- $l_R(U) = U$, $u_R(\emptyset) = \emptyset$, l, u are monotone
- If R is serial: $l_R(H) \subseteq u_R(H)$, $u_R(U) = U$, $l_R(\emptyset) = \emptyset$
- If R is reflexive: $l_R(H) \subseteq H \subseteq u_R(H)$

Similarity relation

Rough sets based on a similarity relation $\mathcal{R}$

- Reflexive
- Symmetric

Similarity $S(x) := \{y \in U : x\mathcal{R}y\}$

$\Rightarrow$ A covering of the universe, not a partition

- $\bigcup_x S(x) = U$
- there can exist objects x and y such that $S(x) \cap S(y) \neq \emptyset$

Similarity relation

Rough sets based on a similarity relation $\mathcal{R}$

- Reflexive
- Symmetric

Similarity $S(x) := \{y \in U : x\mathcal{R}y\}$

$\Rightarrow$ A covering of the universe, not a partition

- $\bigcup_x S(x) = U$
- there can exist objects x and y such that $S(x) \cap S(y) \neq \emptyset$

Similarity relation

Rough sets based on a similarity relation $\mathcal{R}$

- Reflexive
- Symmetric

Similarity $S(x) := \{y \in U : x\mathcal{R}y\}$

$\Rightarrow$ A covering of the universe, not a partition

- $\bigcup_x S(x) = U$
- there can exist objects x and y such that $S(x) \cap S(y) \neq \emptyset$

Similarity: Example 1

$\mathcal{R}$ can represent a distance between objects

- Similar temperature if $|Temp(P1) - Temp(P2)| \leq 0.5$
- P1 similar to P2 if they have (at least) half of the attributes equal

$$\frac{|\{a_i \in Att : F(a_i, P1) = F(a_i, P2)\}|}{|Att|} \geq \frac{1}{2}$$

Similarity: Example 1

$\mathcal{R}$ can represent a distance between objects

- Similar temperature if $|Temp(P1) - Temp(P2)| \leq 0.5$
- P1 similar to P2 if they have (at least) half of the attributes equal

$$\frac{|\{a_i \in Att : F(a_i, P1) = F(a_i, P2)\}|}{|Att|} \geq \frac{1}{2}$$

Similarity: Example 1

$\mathcal{R}$ can represent a distance between objects

- Similar temperature if $|Temp(P1) - Temp(P2)| \leq 0.5$
- P1 similar to P2 if they have (at least) half of the attributes equal

$$\frac{|\{a_i \in Att : F(a_i, P1) = F(a_i, P2)\}|}{|Att|} \geq \frac{1}{2}$$

Similarity: Example 2

- Deal with incomplete information

Patient	Pressure	HA	Temperature	MP	Malattia
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	*	NO
P3	High	no	*	yes	B
P4	*	yes	35–36	no	NO
P5	Normal	*	*	yes	NO

$x\mathcal{R}_D y$ iff $\forall a_i \in D \quad F(x,a_i) = F(y,a_i) \quad$ or $\quad F(x,a_i) = * \quad$ or $\quad F(y,a_i) = *$

Similarity: Example 2

- Deal with incomplete information

Patient	Pressure	HA	Temperature	MP	Malattia
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	*	NO
P3	High	no	*	yes	B
P4	*	yes	35–36	no	NO
P5	Normal	*	*	yes	NO

$x\mathcal{R}_D y$ iff $\forall a_i \in D$ $\quad F(x,a_i) = F(y,a_i)$ or $F(x,a_i) = *$ or $F(y,a_i) = *$

Similarity: Example 2

- Deal with incomplete information

Patient	Pressure	HA	Temperature	MP	Malattia
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	*	NO
P3	High	no	*	yes	B
P4	*	yes	35–36	no	NO
P5	Normal	*	*	yes	NO

$x\mathcal{R}_D y$ iff $\forall a_i \in D$ $\quad F(x,a_i) = F(y,a_i)$ or $F(x,a_i) = *$ or $F(y,a_i) = *$

Outline

Aim

- Simplify the table: eliminate "useless" attributes
- Given a decision system, found the rules:

$$\text{condition attribute} \rightarrow \text{decision}$$

Example:

If Pressure = Normal AND Temp. = 38–39 THEN Disease = A

Aim

- Simplify the table: eliminate "useless" attributes
- Given a decision system, found the rules:

condition attribute $\rightarrow$ decision

Example:

If Pressure = Normal AND Temp. = 38–39 THEN Disease = A

Aim

- Simplify the table: eliminate "useless" attributes
- Given a decision system, found the rules:

condition attribute → decision

Example:

If Pressure = Normal AND Temp. = 38–39 THEN Disease = A

Aim

- Simplify the table: eliminate “useless” attributes
- Given a decision system, found the rules:

condition attribute $\rightarrow$ decision

Example:

If Pressure = Normal AND Temp. = 38–39 THEN Disease = A

Outline

Inf. Table Reduct - example

Patient	Pressure	HA	Temperature	MP
P1	Normal	yes	38–39	yes
P2	High	no	36–37	yes
P3	High	no	36-37	yes
P4	Low	yes	35–36	no
P5	Normal	yes	36–37	yes

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

Inf. Table Reduct - example

Patient	Pressure	HA	[redacted]	MP
P1	Normal	yes	[redacted]	yes
P2	High	no	[redacted]	yes
P3	High	no	[redacted]	yes
P4	Low	yes	[redacted]	no
P5	Normal	yes	[redacted]	yes

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

$\Pi_{Att\setminus\{Temp\}}$ = {P1,P5}, {P2,P3}, {P4}

Inf. Table Reduct - example

Patient	Pressure	[redacted]	Temperature	MP
P1	Normal	[redacted]	38–39	yes
P2	High	[redacted]	36–37	yes
P3	High	[redacted]	36-37	yes
P4	Low	[redacted]	35–36	no
P5	Normal	[redacted]	36–37	yes

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

$\Pi_{Att \setminus \{HA\}} = \Pi_{Att}$

Inf. Table Reduct - example

Patient	Pressure	[redacted]	Temperature	[redacted]
P1	Normal	[redacted]	38–39	[redacted]
P2	High	[redacted]	36–37	[redacted]
P3	High	[redacted]	36-37	[redacted]
P4	Low	[redacted]	35–36	[redacted]
P5	Normal	[redacted]	36–37	[redacted]

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

$\Pi_{Att\setminus\{HA,MP\}} = \Pi_{Att}$

Inf. Table Reduct - example

Patient	[redacted]	[redacted]	Temperature	[redacted]
P1	[redacted]	[redacted]	38–39	[redacted]
P2	[redacted]	[redacted]	36–37	[redacted]
P3	[redacted]	[redacted]	36-37	[redacted]
P4	[redacted]	[redacted]	35–36	[redacted]
P5	[redacted]	[redacted]	36–37	[redacted]

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

Π_{Temp} = {P1}, {P2,P3,P5}, {P4}

Inf. Table Reduct - example

Patient	Pressure	[redacted]
P1	Normal	
P2	High	
P3	High	
P4	Low	
P5	Normal	

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

$\Pi_{Pressure}$ = {P1,P5}, {P2,P3}, {P4}

Inf. Table Reduct - example

Patient	Pressure	[redacted]	Temperature	[redacted]
P1	Normal	[redacted]	38–39	[redacted]
P2	High	[redacted]	36–37	[redacted]
P3	High	[redacted]	36-37	[redacted]
P4	Low	[redacted]	35–36	[redacted]
P5	Normal	[redacted]	36–37	[redacted]

Π_{Att} = {P1}, {P2,P3}, {P4}, {P5}

$\Pi_{Pressure,Temperature} = \Pi_{Att}$

{Pressure, Temperature} is a reduct of *Att*

Inf. Table Reduct - definition

Definition (Reduct)

$A \subseteq B \subseteq Att$

A is a reduct of B if

1. $\Pi_A = \Pi_B$
2. $\nexists C \subset A$ and $\Pi_C = \Pi_B$

$a \in A \subseteq Att$ is indispensable in A if $\Pi_A \neq \Pi_{A \setminus \{a\}}$

CORE= set of indispensable attributes in Att = intersection of all reducts in Att

Inf. Table Reduct - definition

Definition (Reduct)

$A \subseteq B \subseteq Att$

A is a reduct of B if

1. $\Pi_A = \Pi_B$
2. $\nexists C \subset A$ and $\Pi_C = \Pi_B$

$a \in A \subseteq Att$ is indispensable in A if $\Pi_A \neq \Pi_{A \setminus \{a\}}$

CORE= set of indispensable attributes in Att = intersection of all reducts in Att

Inf. Table Reduct - definition

Definition (Reduct)

$A \subseteq B \subseteq Att$
A is a reduct of B if

1. $\Pi_A = \Pi_B$
2. $\nexists C \subset A$ and $\Pi_C = \Pi_B$

$a \in A \subseteq Att$ is indispensable in A if $\Pi_A \neq \Pi_{A \setminus \{a\}}$

CORE= set of indispensable attributes in Att = intersection of all reducts in Att

Inf. Table Reduct - definition

Definition (Reduct)

$A \subseteq B \subseteq Att$

A is a reduct of B if

1. $\Pi_A = \Pi_B$
2. $\nexists C \subset A$ and $\Pi_C = \Pi_B$

$a \in A \subseteq Att$ is indispensable in A if $\Pi_A \neq \Pi_{A \setminus \{a\}}$

CORE= set of indispensable attributes in Att = intersection of all reducts in Att

Inf. Table Reduct - definition

Definition (Reduct)

$A \subseteq B \subseteq Att$

A is a reduct of B if

1. $\Pi_A = \Pi_B$
2. $\nexists C \subset A$ and $\Pi_C = \Pi_B$

$a \in A \subseteq Att$ is indispensable in A if $\Pi_A \neq \Pi_{A \setminus \{a\}}$

CORE= set of indispensable attributes in *Att* = intersection of all reducts in *Att*

Complexity issues

- Given n attributes, there are at most $O(\frac{3^n}{\sqrt{n}})$ reducts
- Find the shortest reduct is a NP^{NP} complete problem
  - reduction to the prime implicant problem by means of the *discernibility matrix*

Solutions

- Heuristics (Approximate reducts, genetic algorithms, entropy, ...)
- Parallel algorithms

Complexity issues

- Given n attributes, there are at most $O(\frac{3^n}{\sqrt{n}})$ reducts
- Find the shortest reduct is a NP^{NP} complete problem
    - reduction to the prime implicant problem by means of the *discernibility matrix*

Solutions

- Heuristics (Approximate reducts, genetic algorithms, entropy, ...)
- Parallel algorithms

Complexity issues

- Given n attributes, there are at most $O(\frac{3^n}{\sqrt{n}})$ reducts
- Find the shortest reduct is a NP^{NP} complete problem
  - reduction to the prime implicant problem by means of the *discernibility matrix*

Solutions

- Heuristics (Approximate reducts, genetic algorithms, entropy, ...)
- Parallel algorithms

Outline

Reduct

Definition (Reduct)

A reduct is a minimal subset of condition $C \subseteq ATT$ that preserves classification wrt the decision attribute

1. Consistence: same ability of the whole ATT to distinguish objects belonging to two different decision classes
2. Minimality: any smaller subset is not consistent

Reduct

Definition (Reduct)

A reduct is a minimal subset of condition $C \subseteq ATT$ that preserves classification wrt the decision attribute

1. Consistence: same ability of the whole ATT to distinguish objects belonging to two different decision classes
2. Minimality: any smaller subset is not consistent

Reduct

Definition (Reduct)

A reduct is a minimal subset of condition $C \subseteq ATT$ that preserves classification wrt the decision attribute

1. Consistence: same ability of the whole ATT to distinguish objects belonging to two different decision classes
2. Minimality: any smaller subset is not consistent

Example

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

Example

Patient	Pressure	HA	[redacted]	MP	Disease
P1	Normal	yes	[redacted]	yes	A
P3	High	no	[redacted]	yes	B
P4	Low	yes	[redacted]	no	NO
P5	Normal	yes	[redacted]	yes	NO

Example

Patient	Pressure	HA	[redacted]	MP	Disease
P1	Normal	yes	[redacted]	yes	A
P3	High	no	[redacted]	yes	B
P4	Low	yes	[redacted]	no	NO
P5	Normal	yes	[redacted]	yes	NO

Example

Patient	Pressure	[redacted]	Temperature	MP	Disease
P1	Normal	[redacted]	38–39	yes	A
P3	High	[redacted]	36-37	yes	B
P4	Low	[redacted]	35–36	no	NO
P5	Normal	[redacted]	36–37	yes	NO

Example

Patient	Pressure	[redacted]	Temperature	[redacted]	Disease
P1	Normal	[redacted]	38–39	[redacted]	A
P3	High	[redacted]	36-37	[redacted]	B
P4	Low	[redacted]	35–36	[redacted]	NO
P5	Normal	[redacted]	36–37	[redacted]	NO

Example

Patient	[redacted]	[redacted]	Temperature	[redacted]	Disease
P1	[redacted]	[redacted]	38–39	[redacted]	A
P3	[redacted]	[redacted]	36-37	[redacted]	B
P4	[redacted]	[redacted]	35–36	[redacted]	NO
P5	[redacted]	[redacted]	36–37	[redacted]	NO

Example

Patient	Pressure	[redacted]	Disease
P1	Normal	[redacted]	A
P3	High	[redacted]	B
P4	Low	[redacted]	NO
P5	Normal	[redacted]	NO

Example: rules

Patient	Pressure	[redacted]	Temperature	[redacted]	Disease
P1	Normal	[redacted]	38–39	[redacted]	A
P3	High	[redacted]	36-37	[redacted]	B
P4	Low	[redacted]	35–36	[redacted]	NO
P5	Normal	[redacted]	36–37	[redacted]	NO

Reduct = {Pressure, Temperature}

IF Pressure = Normal AND Temp. = 38–39 THEN Disease = A
IF Pressure = High AND Temp. = 36–37 THEN Disease = B
IF Pressure = Low AND Temp. = 35–36 THEN Disease = NO
IF Pressure = Normal AND Temp. = 36–37 THEN Disease = NO

Example: rules

Patient	Pressure	[redacted]	Temperature	[redacted]	Disease
P1	Normal	[redacted]	38–39	[redacted]	A
P3	High	[redacted]	36-37	[redacted]	B
P4	Low	[redacted]	35–36	[redacted]	NO
P5	Normal	[redacted]	36–37	[redacted]	NO

Reduct = {Pressure, Temperature}

IF Pressure = Normal AND Temp. = 38–39 THEN Disease = A
IF Pressure = High AND Temp. = 36–37 THEN Disease = B
IF Pressure = Low AND Temp. = 35–36 THEN Disease = NO
IF Pressure = Normal AND Temp. = 36–37 THEN Disease = NO

Example: rules

Patient	Pressure	[redacted]	Temperature	[redacted]	Disease
P1	Normal	[redacted]	38–39	[redacted]	A
P3	High	[redacted]	36-37	[redacted]	B
P4	Low	[redacted]	35–36	[redacted]	NO
P5	Normal	[redacted]	36–37	[redacted]	NO

Reduct = {Pressure, Temperature}

IF Pressure = Normal AND Temp. = 38–39 THEN Disease = A
IF Pressure = High AND Temp. = 36–37 THEN Disease = B
IF Pressure = Low AND Temp. = 35–36 THEN Disease = NO
IF Pressure = Normal AND Temp. = 36–37 THEN Disease = NO

Example: rules

Patient	Pressure	[redacted]	Temperature	[redacted]	Disease
P1	Normal	[redacted]	38–39	[redacted]	A
P3	High	[redacted]	36-37	[redacted]	B
P4	Low	[redacted]	35–36	[redacted]	NO
P5	Normal	[redacted]	36–37	[redacted]	NO

Reduct = {Pressure, Temperature}

IF Pressure = Normal AND Temp. = 38–39 THEN Disease = A
IF Pressure = High AND Temp. = 36–37 THEN Disease = B
IF Pressure = Low AND Temp. = 35–36 THEN Disease = NO
IF Pressure = Normal AND Temp. = 36–37 THEN Disease = NO

Example: rules

Patient	Pressure	[redacted]	Temperature	[redacted]	Disease
P1	Normal	[redacted]	38–39	[redacted]	A
P3	High	[redacted]	36-37	[redacted]	B
P4	Low	[redacted]	35–36	[redacted]	NO
P5	Normal	[redacted]	36–37	[redacted]	NO

Reduct = {Pressure, Temperature}

IF Pressure = Normal AND Temp. = 38–39 THEN Disease = A
IF Pressure = High AND Temp. = 36–37 THEN Disease = B
IF Pressure = Low AND Temp. = 35–36 THEN Disease = NO
IF Pressure = Normal AND Temp. = 36–37 THEN Disease = NO

Outline

Solution 1: Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A\, y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Solution 1: Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A\, y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Solution 1: Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A\, y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Solution 1: Generalized Decision

Patient	Pressure	HA	Temperature	MP	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

- Generalized decision: $\delta_A : U \to \mathcal{P}(Val)$
- Example: $\delta_{ATT}(P2) = \{NO, B\}$
- Definition:

$$\delta_A(x) = \{i \in Val : \exists y, x\, I_A y \text{ and } F(y,d) = i\}$$

- If $\forall x \in U : |\delta_A(x)| = 1$ then the system is consistent

Generalized Decision reduct

Definition

Given a set of attributes $A \subseteq B \subseteq ATT$, A is a reduct of B if

- $\delta_A = \delta_B$ (I do not introduce further inconsistency)
- Minimality: $\nexists C \subset A$ such that $\delta_C = \delta_B$

Generalized Decision Reduct - example

Patient	Pressure	HA	Temperature	MP	Disease	δ_{Att}
P1	Normal	yes	38–39	yes	A	A,
P2	High	no	36–37	yes	NO	B,NO
P3	High	no	36-37	yes	B	B,NO
P4	Low	yes	35–36	no	NO	NO
P5	Normal	yes	36–37	yes	NO	B

- Reduct {*Pressure*, *Temperature*}
- If (Pressure =High) AND (Temp=36-37) THEN (Disease = NO) OR (Disease = B)

Generalized Decision Reduct - example

Patient	Pressure	HA	Temperature	MP	Disease	δ_{Att}
P1	Normal	yes	38–39	yes	A	A,
P2	High	no	36–37	yes	NO	B,NO
P3	High	no	36-37	yes	B	B,NO
P4	Low	yes	35–36	no	NO	NO
P5	Normal	yes	36–37	yes	NO	B

- Reduct $\{Pressure, Temperature\}$
- If (Pressure =High) AND (Temp=36-37) THEN (Disease = NO) OR (Disease = B)

Generalized Decision Reduct - example

Patient	Pressure	HA	Temperature	MP	Disease	δ_{Att}
P1	Normal	yes	38–39	yes	A	A,
P2	High	no	36–37	yes	NO	B,NO
P3	High	no	36-37	yes	B	B,NO
P4	Low	yes	35–36	no	NO	NO
P5	Normal	yes	36–37	yes	NO	B

- Reduct $\{Pressure, Temperature\}$
- If (Pressure =High) AND (Temp=36-37) THEN (Disease = NO) OR (Disease = B)

Solution 2: Dependence

Definition

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq Att$ a set of attributes, X_i the decision classes
The Coefficient of Dependence of decision d from A is

$$Dip(A, d) = \frac{\sum |L_A(X_i)|}{|X|}$$

Dip(A,d) is the ratio of correctly classified objects by the set of attributes A

Dip(A,d) = 1 if the system is consistent

Solution 2: Dependence

Definition

Let $\mathcal{S}(U)$ be a decision system

$A \subseteq Att$ a set of attributes, X_i the decision classes

The Coefficient of Dependence of decision d from A is

$$Dip(A,d) = \frac{\sum |L_A(X_i)|}{|X|}$$

Dip(A,d) is the ratio of correctly classified objects by the set of attributes A

Dip(A,d) = 1 if the system is consistent

Solution 2: Dependence

Definition

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq Att$ a set of attributes, X_i the decision classes
The Coefficient of Dependence of decision d from A is

$$Dip(A,d) = \frac{\sum |L_A(X_i)|}{|X|}$$

Dip(A,d) is the ratio of correctly classified objects by the set of attributes A

Dip(A,d) = 1 if the system is consistent

Solution 2: Dependence

Definition

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq Att$ a set of attributes, X_i the decision classes
The Coefficient of Dependence of decision d from A is

$$Dip(A, d) = \frac{\sum |L_A(X_i)|}{|X|}$$

Dip(A,d) is the ratio of correctly classified objects by the set of attributes A

Dip(A,d) = 1 if the system is consistent

Solution 2: Dependence

Definition

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq Att$ a set of attributes, X_i the decision classes
The Coefficient of Dependence of decision d from A is

$$Dip(A, d) = \frac{\sum |L_A(X_i)|}{|X|}$$

Dip(A,d) is the ratio of correctly classified objects by the set of attributes A

Dip(A,d) = 1 if the system is consistent

Reduct: dependence definition

Definition (Reduct)

Let $\mathcal{S}(U)$ be a decision system

$A \subseteq B \subseteq Att$, A is a reduct of B if

1. Dip(A,d)=Dip(B,d)
2. Minimality: $\nexists C \subset A$ such that $Dip(C, d) = Dip(B, d)$

Reduct: dependence definition

Definition (Reduct)

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq B \subseteq Att$, A is a reduct of B if

1. Dip(A,d)=Dip(B,d)
2. Minimality: $\nexists C \subset A$ such that $Dip(C,d) = Dip(B,d)$

Reduct: dependence definition

Definition (Reduct)

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq B \subseteq Att$, A is a reduct of B if

1. Dip(A,d)=Dip(B,d)
2. Minimality: $\nexists C \subset A$ such that $Dip(C, d) = Dip(B, d)$

Reduct: dependence definition

Definition (Reduct)

Let $\mathcal{S}(U)$ be a decision system
$A \subseteq B \subseteq Att$, A is a reduct of B if

1. Dip(A,d)=Dip(B,d)
2. Minimality: $\nexists C \subset A$ such that $Dip(C, d) = Dip(B, d)$

Reduct: dependence example

Patient	Pressure	HA	Temperature	DM	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

$L_C(X_A) = \{P1\}$, $L_C(X_{NO}) = \{P4, P5\}$, $L_C(X_B) = \emptyset$

$Dip(Att, Disease) = \frac{3}{5}$

$Dip(\{Pressure, Temperature, DM\}, Disease) = \frac{3}{5}$

$Dip(\{Pressure, Temperature\}, Disease) = \frac{3}{5}$

IF (Pressure=High AND Temp= 36–37) THEN (Disease =NO OR Disease =B)

Reduct: dependence example

Patient	Pressure	HA	Temperature	DM	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

$L_C(X_A) = \{P1\}$, $L_C(X_{NO}) = \{P4, P5\}$, $L_C(X_B) = \emptyset$

$Dip(Att, Disease) = \frac{3}{5}$

$Dip(\{Pressure, Temperature, DM\}, Disease) = \frac{3}{5}$

$Dip(\{Pressure, Temperature\}, Disease) = \frac{3}{5}$

IF (Pressure=High AND Temp= 36–37) THEN (Disease =NO OR Disease =B)

Reduct: dependence example

Patient	Pressure	HA	Temperature	DM	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

$L_C(X_A) = \{P1\}$, $L_C(X_{NO}) = \{P4, P5\}$, $L_C(X_B) = \emptyset$

$Dip(Att, Disease) = \frac{3}{5}$

$Dip(\{Pressure, Temperature, DM\}, Disease) = \frac{3}{5}$

$Dip(\{Pressure, Temperature\}, Disease) = \frac{3}{5}$

IF (Pressure=High AND Temp= 36–37) THEN (Disease =NO OR Disease =B)

Reduct: dependence example

Patient	Pressure	HA	Temperature	DM	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

$L_C(X_A) = \{P1\}$, $L_C(X_{NO}) = \{P4, P5\}$, $L_C(X_B) = \emptyset$

$Dip(Att, Disease) = \frac{3}{5}$
$Dip(\{Pressure, Temperature, DM\}, Disease) = \frac{3}{5}$
$Dip(\{Pressure, Temperature\}, Disease) = \frac{3}{5}$

IF (Pressure=High AND Temp= 36–37) THEN (Disease =NO OR Disease =B)

Reduct: dependence example

Patient	Pressure	HA	Temperature	DM	Disease
P1	Normal	yes	38–39	yes	A
P2	High	no	36–37	yes	NO
P3	High	no	36-37	yes	B
P4	Low	yes	35–36	no	NO
P5	Normal	yes	36–37	yes	NO

$L_C(X_A) = \{P1\}$, $L_C(X_{NO}) = \{P4, P5\}$, $L_C(X_B) = \emptyset$

$Dip(Att, Disease) = \frac{3}{5}$
$Dip(\{Pressure, Temperature, DM\}, Disease) = \frac{3}{5}$
$Dip(\{Pressure, Temperature\}, Disease) = \frac{3}{5}$

IF (Pressure=High AND Temp= 36–37) THEN (Disease =NO OR Disease =B)

Software

Free software based on rough sets

- Rosetta (2001), limited to tables with 500 objects and 20 attributes
  http://www.lcb.uu.se/tools/rosetta
- Rough Set and Machine Learning Open Source in Java (2019)
  Also avalaible in WEKA
  https://rseslib.mimuw.edu.pl/index.html
- R package "RoughSets: Data Analysis Using Rough Set and Fuzzy Rough Set Theories" (2019) https://cran.r-project.org/web/packages/RoughSets/index.html
- R package "Soft Clustering" (2019) https://cran.r-project.org/web/packages/SoftClustering/index.html
- Fuzzy Rough Learn (2021) python library
  https://fuzzy-rough-learn.readthedocs.io/en/latest/

Software

Free software based on rough sets

- Rosetta (2001), limited to tables with 500 objects and 20 attributes
  `http://www.lcb.uu.se/tools/rosetta`
- Rough Set and Machine Learning Open Source in Java (2019)
  Also avalaible in WEKA
  `https://rseslib.mimuw.edu.pl/index.html`
- R package "RoughSets: Data Analysis Using Rough Set and Fuzzy Rough Set Theories" (2019) `https://cran.r-project.org/web/packages/RoughSets/index.html`
- R package "Soft Clustering" (2019) `https://cran.r-project.org/web/packages/SoftClustering/index.html`
- Fuzzy Rough Learn (2021) python library
  `https://fuzzy-rough-learn.readthedocs.io/en/latest/`

Software

Free software based on rough sets

- Rosetta (2001), limited to tables with 500 objects and 20 attributes
  `http://www.lcb.uu.se/tools/rosetta`
- Rough Set and Machine Learning Open Source in Java (2019)
  Also avalaible in WEKA
  `https://rseslib.mimuw.edu.pl/index.html`
- R package "RoughSets: Data Analysis Using Rough Set and Fuzzy Rough Set Theories" (2019) `https://cran.r-project.org/web/packages/RoughSets/index.html`
- R package "Soft Clustering" (2019) `https://cran.r-project.org/web/packages/SoftClustering/index.html`
- Fuzzy Rough Learn (2021) python library
  `https://fuzzy-rough-learn.readthedocs.io/en/latest/`