# ROUGH SET BASED DECISION SUPPORT

Roman Słowiński

Laboratory of Intelligent Decision Support Systems
Institute of Computing Science
Poznań University of Technology

© Roman Słowiński

# Motywacje

- Wzrasta przepaść między generowaniem danych a ich zrozumieniem
- Odkrywanie wiedzy z danych (ang. *Knowledge Discovery, Data Mining*)
- OWD jest procesem identyfikowania przez indukcję:
  - prawdziwych,
  - nietrywialnych,
  - potencjalnie użytecznych,
  - bezpośrednio zrozumiałych

  **wzorców** w danych
- Wzorzec = reguła, trend, zjawisko, prawidłowość, anomalia, hipoteza, funkcja, itp.
- Wzorce są użyteczne dla wyjaśniania sytuacji opisanych przez dane oraz do predykcji przyszłych sytuacji

# Motywacje – odkrywanie wzorców

- <u>Przykład danych diagnostycznych</u>
- 176 autobusów (obiektów)
- 8 symptomów (atrybutów)
- Globalny stan techniczny:
  - **3** – dobry (użytkować)
  - **2** – do przeglądu
  - **1** – do remontu (wycofać)
- Odkryć wzorce = znaleźć zależności między symptomami a stanem technicznym
- Wzorce wyjaśniają decyzje eksperta i wspomagają diagnozowanie nowych przypadków

Examples:

| | MaxSpeed | ComprPressure | Blacking | Torque | SummerCons | WinterCons | OilCons | HorsePower | State |
|---|---|---|---|---|---|---|---|---|---|
| 1. | 90 | 2 | 38 | 481 | 21 | 26 | 0 | 145 | 3 |
| 2. | 76 | 2 | 70 | 420 | 22 | 25 | 2 | 110 | 1 |
| 3. | 63 | 1 | 82 | 400 | 22 | 24 | 3 | 101 | 1 |
| 4. | 90 | 2 | 49 | 477 | 21 | 25 | 1 | 138 | 3 |
| 5. | 85 | 2 | 52 | 460 | 21 | 25 | 1 | 130 | 2 |
| 6. | 72 | 2 | 73 | 425 | 23 | 27 | 2 | 112 | 1 |
| 7. | 88 | 2 | 50 | 480 | 21 | 24 | 1 | 140 | 3 |
| 8. | 87 | 2 | 56 | 465 | 22 | 27 | 1 | 135 | 3 |
| 9. | 90 | 2 | 16 | 486 | 26 | 27 | 0 | 150 | 3 |
| 10. | 60 | 1 | 95 | 400 | 23 | 24 | 4 | 96 | 1 |
| 11. | 80 | 2 | 60 | 451 | 21 | 26 | 1 | 125 | 1 |
| 12. | 78 | 2 | 63 | 448 | 21 | 26 | 1 | 120 | 2 |
| 13. | 90 | 2 | 26 | 482 | 22 | 24 | 0 | 148 | 3 |
| 14. | 62 | 1 | 93 | 400 | 22 | 28 | 3 | 100 | 1 |
| 15. | 82 | 2 | 54 | 461 | 22 | 26 | 1 | 132 | 2 |
| 16. | 65 | 2 | 67 | 402 | 22 | 23 | 2 | 103 | 1 |
| 17. | 90 | 2 | 51 | 468 | 22 | 26 | 1 | 138 | 3 |
| 18. | 90 | 2 | 15 | 488 | 20 | 23 | 0 | 150 | 3 |
| 19. | 76 | 2 | 65 | 428 | 27 | 33 | 2 | 116 | 1 |
| 20. | 85 | 2 | 50 | 454 | 21 | 26 | 1 | 129 | 2 |
| 21. | 85 | 2 | 58 | 450 | 22 | 25 | 1 | 126 | 2 |
| 22. | 88 | 2 | 48 | 458 | 22 | 25 | 1 | 130 | 3 |
| 23. | 60 | 1 | 90 | 400 | 24 | 28 | 4 | 95 | 1 |
| 24. | 64 | 2 | 71 | 420 | 23 | 25 | 2 | 105 | 1 |
| 25. | 75 | 2 | 64 | 432 | 22 | 25 | 1 | 114 | 2 |
| 26. | 74 | 2 | 64 | 420 | 21 | 25 | 1 | 110 | 2 |
| 27. | 68 | 2 | 70 | 400 | 22 | 26 | 2 | 100 | 1 |

Attributes: 9 of 10 | Examples: 76 | Decision: State | Missing Values: No

# Introduction

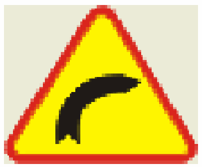
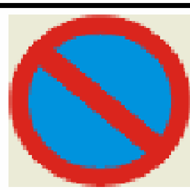
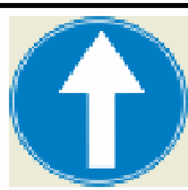
- Discovering knowledge from data means to find concise classification patterns that agree with *situations* described by the data
- The situations are described by a set of *attributes*, called also properties, features, characteristics, etc.
- The attributes may be either on *condition* or *decision* side of the description, corresponding to input or output of a decision situation
- The situations may be objects, states, examples, etc.
  It will be convenient to call them *objects* from now
- We present a knowledge discovery method for multiattribute decision making, based on the concept of *rough sets* (Pawlak 1982)
- *Rough set theory* provides a framework for dealing with inconsistent or ambiguous data in knowledge discovery
- As the inconsistency and ambiguity follow from information granulation, the core idea of rough sets is that of *granular approximation*

# Classical Rough Set Approach (CRSA)

***Example***

- Classification of basic traffic signs
- There exist three main classes of traffic signs corresponding to:
  - warning (W),
  - interdiction (I),
  - order (O).
- These classes may be distinguished by such attributes as the shape (S) and the principal color (PC) of the sign
- Finally, we give few examples of traffic signs

# CRSA – example of traffic signs

| Traffic sign | Shape (S) | Primary Color (PC) | Class |
|---|---|---|---|
| *a*)  | triangle | yellow | W |
| *b*)   | circle | white | I |
| *c*)  | circle | blue | I |
| *d*)  | circle | blue | O |

# CRSA – example of traffic signs

| Traffic sign | Shape (S) | Primary Color (PC) | Class |
|---|---|---|---|
| *a*)  | triangle | yellow | W |
| *b*)   | circle | white | I |
| *c*)  | circle | blue | I |
| *d*)  | circle | blue | O |

- Granules of knowledge:

  W={*a*}$_{\text{Class}}$, I={*b*,*c*}$_{\text{Class}}$, O={*d*}$_{\text{Class}}$

  {*a*}$_{\text{S,PC}}$, {*b*}$_{\text{S,PC}}$, {*c*,*d*}$_{\text{S,PC}}$

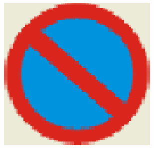
- Explanation of classification in terms of granules generated by S and PC
  - class W includes sign *a* certainly and no other sign possibly
  - class I includes sign *b* certainly and signs *b*, *c* and *d* possibly
  - class O includes no sign certainly and signs *c* and *d* possibly

- *Lower* and *upper approximation* of the classes by attributes S and PC:
  - lower_appx.$_{\text{S,PC}}$(W)={*a*}, upper_appx.$_{\text{S,PC}}$(W)={*a*}
  - lower_appx.$_{\text{S,PC}}$(I)={*b*}, upper_appx.$_{\text{S,PC}}$(I)={*b*,*c*,*d*}
  - lower_appx.$_{\text{S,PC}}$(O)=∅, upper_appx.$_{\text{S,PC}}$(O)={*c*,*d*}
  - boundary$_{\text{S,PC}}$(I)=upper_appx.$_{\text{S,PC}}$(I) – lower_appx.$_{\text{S,PC}}$(I)={*c*,*d*}
  - boundary$_{\text{S,PC}}$(O)=upper_appx.$_{\text{S,PC}}$(O) – lower_appx.$_{\text{S,PC}}$(O)={*c*,*d*}

- The *quality of approximation*: 2/4

# CRSA – example of traffic signs

- To increase the quality of approximation (decrease the ambiguity) we add a new attribute – secondary color (SC)

| Traffic sign | Shape (S) | Primary Color (PC) | Secondary color (SC) | Class |
|---|---|---|---|---|
| *a*)  | triangle | yellow | red | W |
| *b*)  | circle | white | red | I |
| *c*)  | circle | blue | red | I |
| *d*)  | circle | blue | white | O |

- The granules: $\{a\}_{S,PC,SC}$, $\{b\}_{S,PC,SC}$, $\{c\}_{S,PC,SC}$, $\{d\}_{S,PC,SC}$
- Quality of approximation: 4/4=1

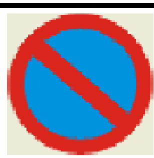
# CRSA – example of traffic signs

- Are all three attributes necessary to characterize precisely the classes W, I, O ?

| Traffic sign | Shape (S) | Primary Color (PC) | Secondary color (SC) | Class |
|---|---|---|---|---|
| *a*)  | triangle | yellow | red | W |
| *b*)  | circle | white | red | I |
| *c*)  | circle | blue | red | I |
| *d*)  | circle | blue | white | O |

- The granules: $\{a\}_{PC,SC}$, $\{b\}_{PC,SC}$, $\{c\}_{PC,SC}$, $\{d\}_{PC,SC}$
- Quality of approximation: 4/4=1

# CRSA – example of traffic signs

| Traffic sign | Shape (S) | Primary Color (PC) | Secondary color (SC) | Class |
|---|---|---|---|---|
| *a*)  | triangle | yellow | red | W |
| *b*)   | circle | white | red | I |
| *c*)  | circle | blue | red | I |
| *d*)  | circle | blue | white | O |

- The granules: $\{a\}_{S,SC}$, $\{b,c\}_{S,SC}$, $\{d\}_{S,SC}$
- *Reducts* of the set of attributes: {PC, SC} and {S, SC}
- Intersection of reducts is the *core*: {SC}

# CRSA – example of traffic signs

- The minimal representation of knowledge contained in the Table – *decision rules*

| Traffic sign | | Shape (S) | Primary Color (PC) | Secondary color (SC) | Class |
|---|---|---|---|---|---|
| *a*) |  | triangle | yellow | red | W |
| *b*) |  | circle | white | red | I |
| *c*) |  | circle | blue | red | I |
| *d*) |  | circle | blue | white | O |

rule #1: *if* S=triangle, *then* Class=W {*a*}
rule #2: *if* S=circle *and* SC=red, *then* Class=I {*b*,*c*}
rule #3: *if* SC=white, *then* Class=O {*d*}

- Decision rules are *classification patterns* discovered from data contained in the table

# CRSA – example of traffic signs

- Alternative set of decision rules

| Traffic sign | | Shape (S) | Primary Color (PC) | Secondary color (SC) | Class |
|---|---|---|---|---|---|
| *a*) |  | ~~triangle~~ | yellow | ~~red~~ | W |
| *b*) |  | ~~circle~~ | white | ~~red~~ | I |
| *c*) |  | ~~circle~~ | blue | red | I |
| *d*) |  | ~~circle~~ | ~~blue~~ | white | O |

rule #1’: *if* PC=yellow, *then* Class=W {*a*}

rule #2’: *if* PC=white, *then* Class=I {*b* }

rule #3’: *if* PC=blue *and* SC=red, *then* Class=I {*c*}

rule #4’: *if* SC=white, *then* Class=O {*d*}

# CRSA – example of traffic signs

- Decision rules induced from the original table

| Traffic sign | Shape (S) | Primary Color (PC) | Class |
|---|---|---|---|
| *a*)  | triangle | ~~yellow~~ | W |
| *b*)  | ~~circle~~ | white | I |
| *c*)  | ~~circle~~ | blue | I |
| *d*)  | ~~circle~~ | blue | O |

rule #1'': *if* S=triangle, *then* Class=W $\{a\}$
rule #2'': *if* PC=white, *then* Class=I $\{b\}$
rule #3'': *if* PC=blue, *then* Class=I or O $\{c,d\}$

- Rules #1'' & #2'' – certain rules induced from lower approximations of W and I
- Rule #3'' – approximate rule induced from the boundary of I and O

# CRSA – example of traffic signs

- Useful results:
  - a characterization of decision classes (even in case of inconsistency) in terms of chosen attributes by lower and upper approximation,
  - a measure of the quality of approximation indicating how good the chosen set of attributes is for approximation of the classification,
  - reduction of knowledge contained in the table to the description by relevant attributes belonging to reducts,
  - the core of attributes indicating indispensable attributes,
  - decision rules induced from lower and upper approximations of decision classes show classification patterns existing in data.

# CRSA – formal definitions

- Approximation space

  $U$ = finite set of objects (universe)

  $C$ = set of *condition attributes*

  $D$ = set of *decision attributes*

  $C \cap D = \emptyset$

$$X_C = \prod_{q=1}^{|C|} X_q$$ – condition attribute space

$$X_D = \prod_{q=1}^{|D|} X_q$$ – decision attribute space

# CRSA – formal definitions

- Indiscernibility relation in the approximation space

  ***x* is indiscernible with *y*** by $P \subseteq C$ in $X_P$ iff $x_q = y_q$ for all $q \in P$

  ***x* is indiscernible with *y*** by $R \subseteq D$ in $X_D$ iff $x_q = y_q$ for all $q \in R$

  $I_P(x)$, $I_R(x)$ – equivalence classes including $x$

  $I_D$ makes a partition of $U$ into decision classes $\mathbf{Cl} = \{Cl_t,\ t = 1, \ldots, n\}$

- Granules of knowledge are bounded sets:

  $I_P(x)$ in $X_P$ and $I_R(x)$ in $X_R$ ($P \subseteq C$ and $R \subseteq D$)

- Classification patterns to be discovered are functions representing granules $I_R(x)$ by granules $I_P(x)$

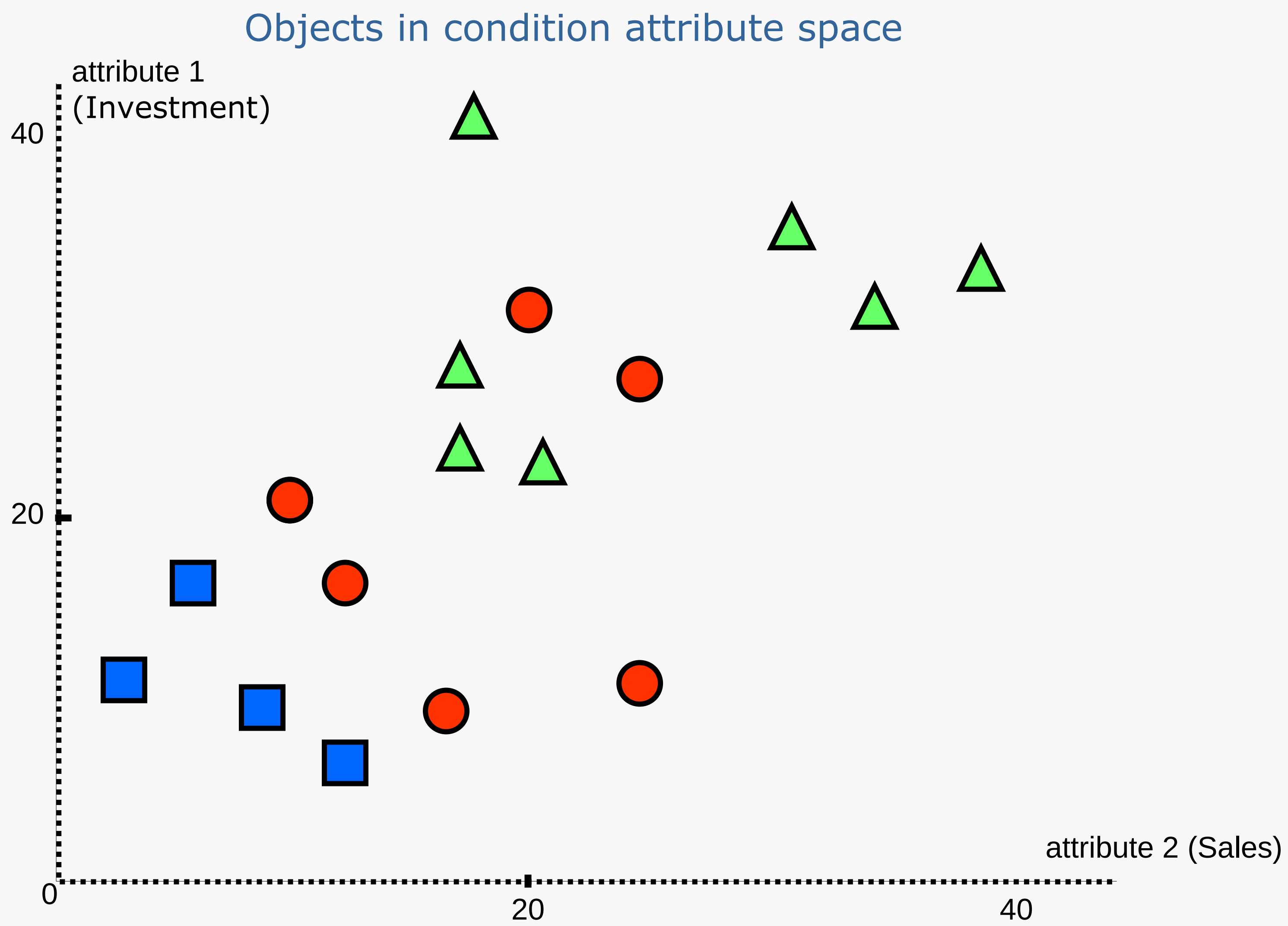
# CRSA – illustration of formal definitions

- Example

Objects = firms

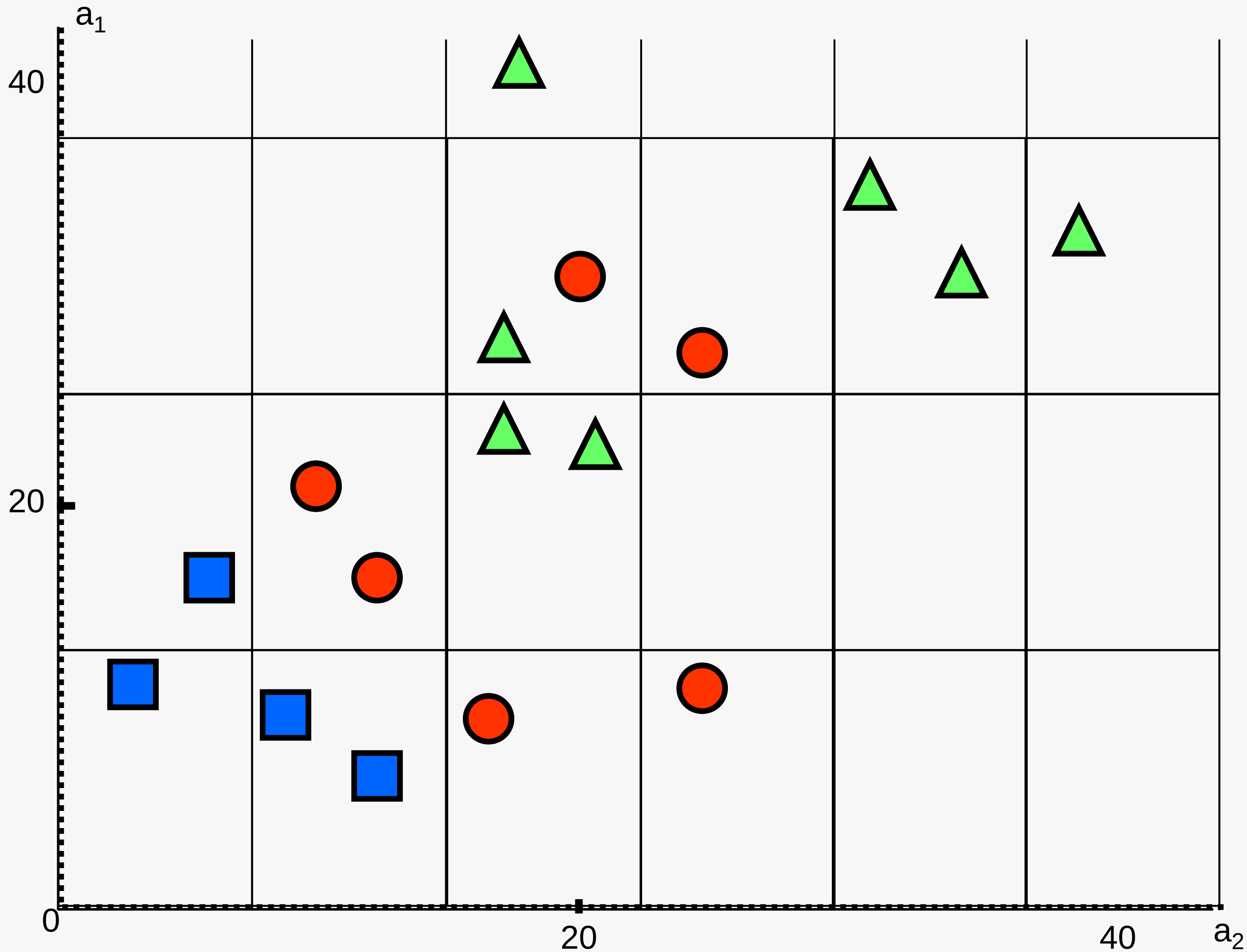
| Investments | Sales | Effectiveness |
|---|---|---|
| 40 | 17,8 | ▲ High |
| 35 | 30 | ▲ High |
| 32.5 | 39 | ▲ High |
| 31 | 35 | ▲ High |
| 27.5 | 17.5 | ▲ High |
| 24 | 17.5 | ▲ High |
| 22.5 | 20 | ▲ High |
| 30.8 | 19 | ● Medium |
| 27 | 25 | ● Medium |
| 21 | 9.5 | ● Medium |
| 18 | 12.5 | ● Medium |
| 10.5 | 25.5 | ● Medium |
| 9.75 | 17 | ● Medium |
| 17.5 | 5 | ■ Low |
| 11 | 2 | ■ Low |
| 10 | 9 | ■ Low |
| 5 | 13 | ■ Low |

# CRSA – illustration of formal definitions



# CRSA – illustration of formal definitions

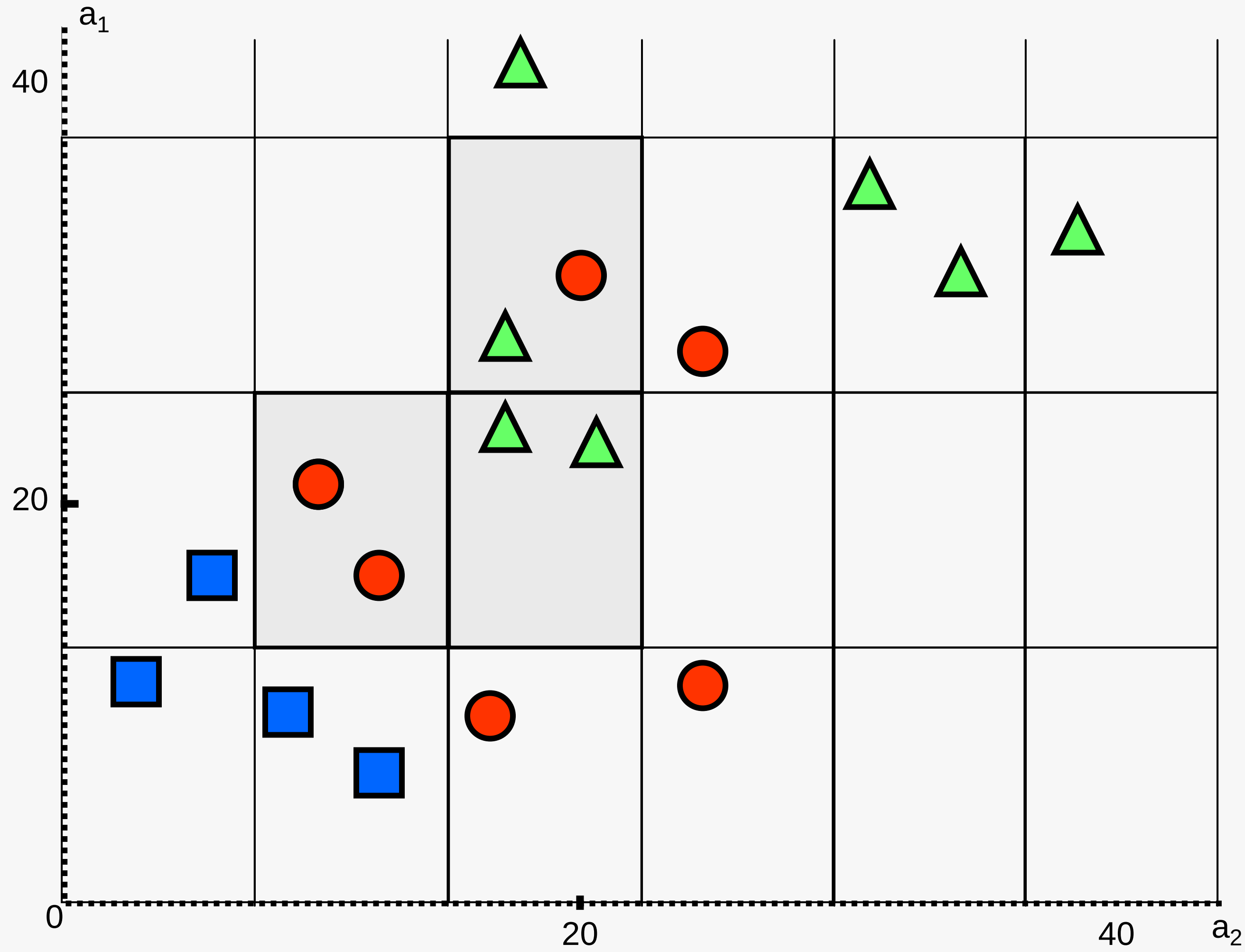
Indiscernibility sets




Quantitative attributes are discretized according to perception of the user

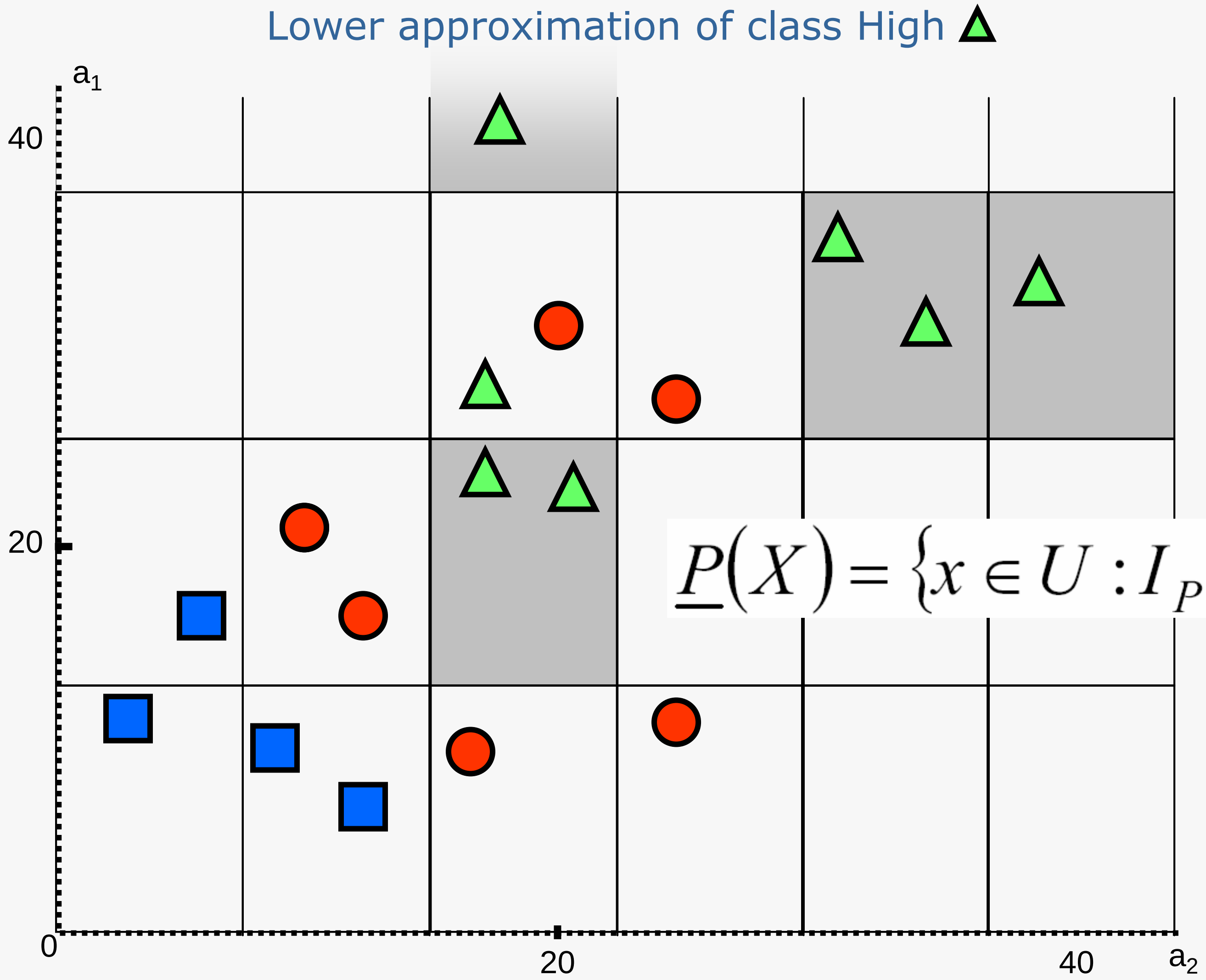
# CRSA – illustration of formal definitions

Granules of knowlegde are bounded sets $I_P(x)$



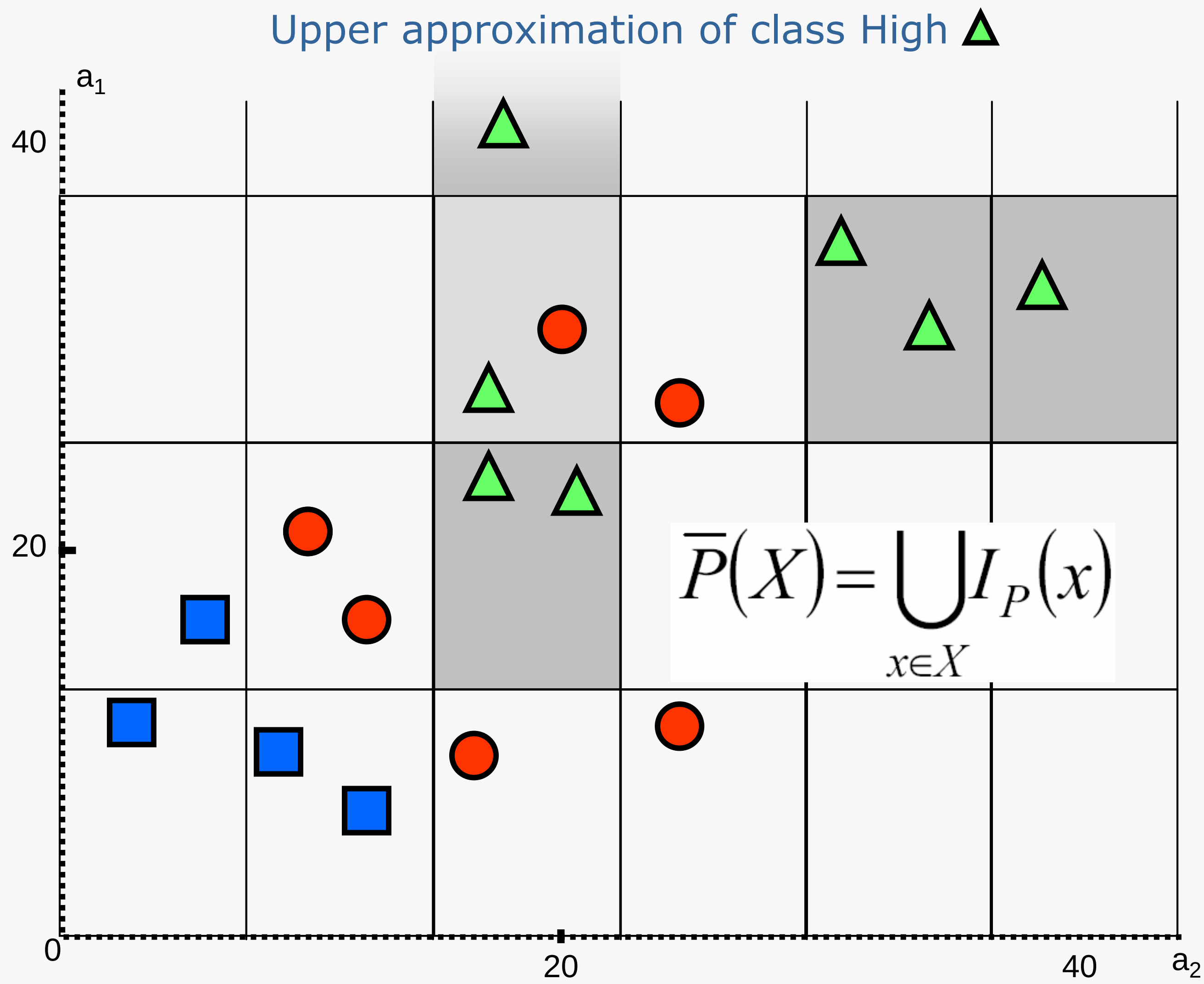
# CRSA – illustration of formal definitions

Lower approximation of class High




$$\underline{P}(X) = \{x \in U : I_P(x) \subseteq X\}$$

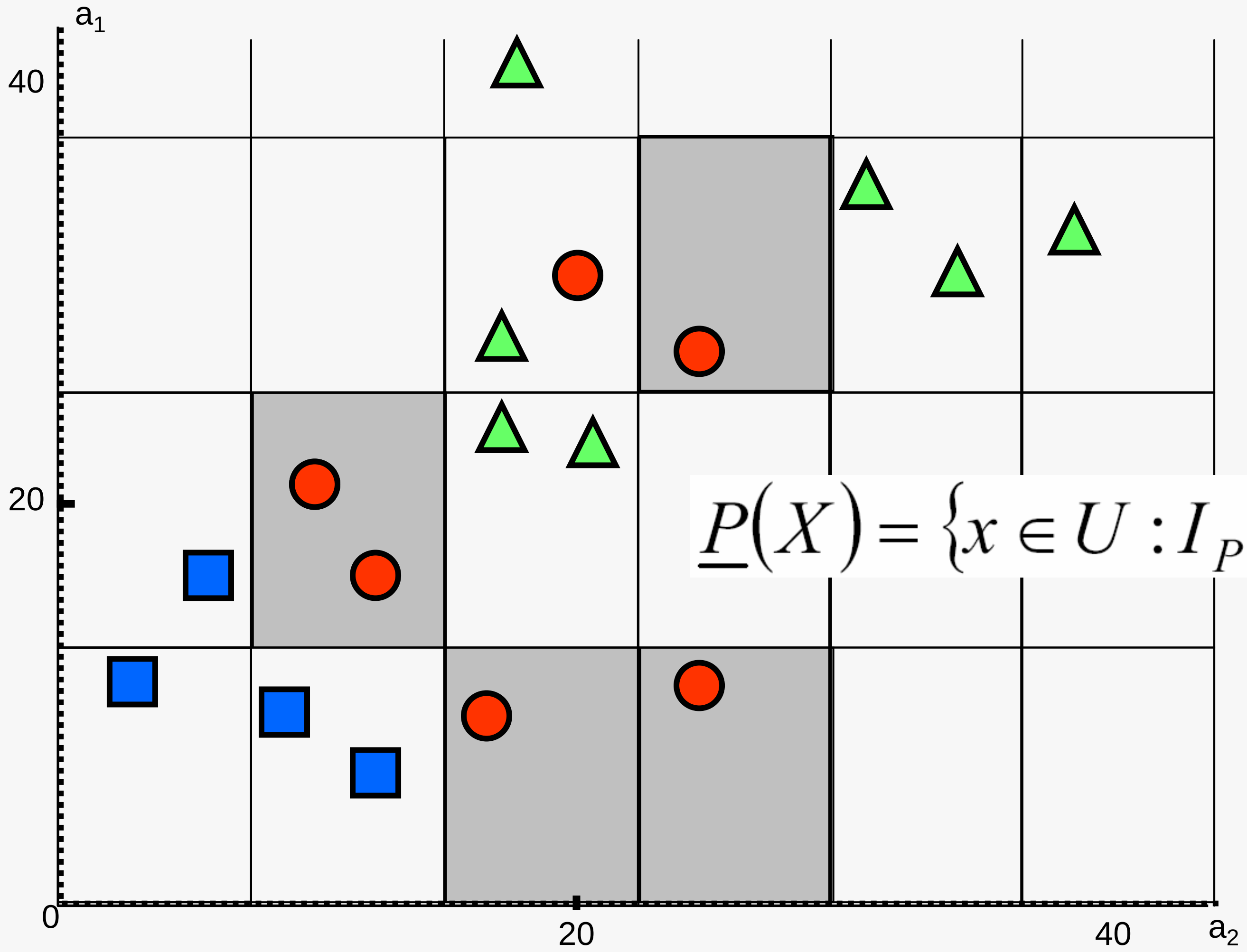
# CRSA – illustration of formal definitions




$$\overline{P}(X)=\bigcup_{x\in X} I_P(x)$$

# CRSA – illustration of formal definitions

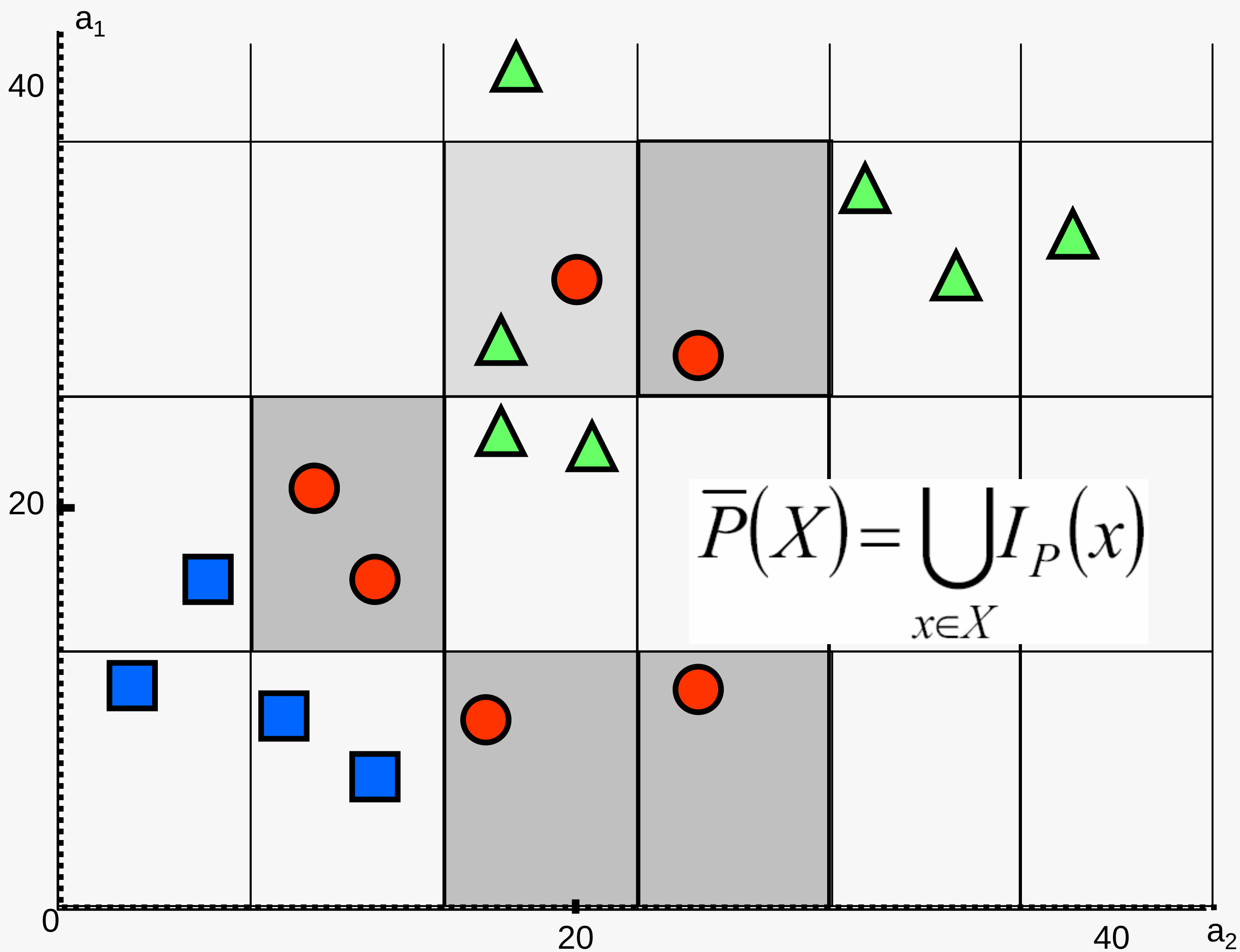
Lower approximation of class Medium




$$\underline{P}(X)=\{x\in U: I_P(x)\subseteq X\}$$

# CRSA – illustration of formal definitions

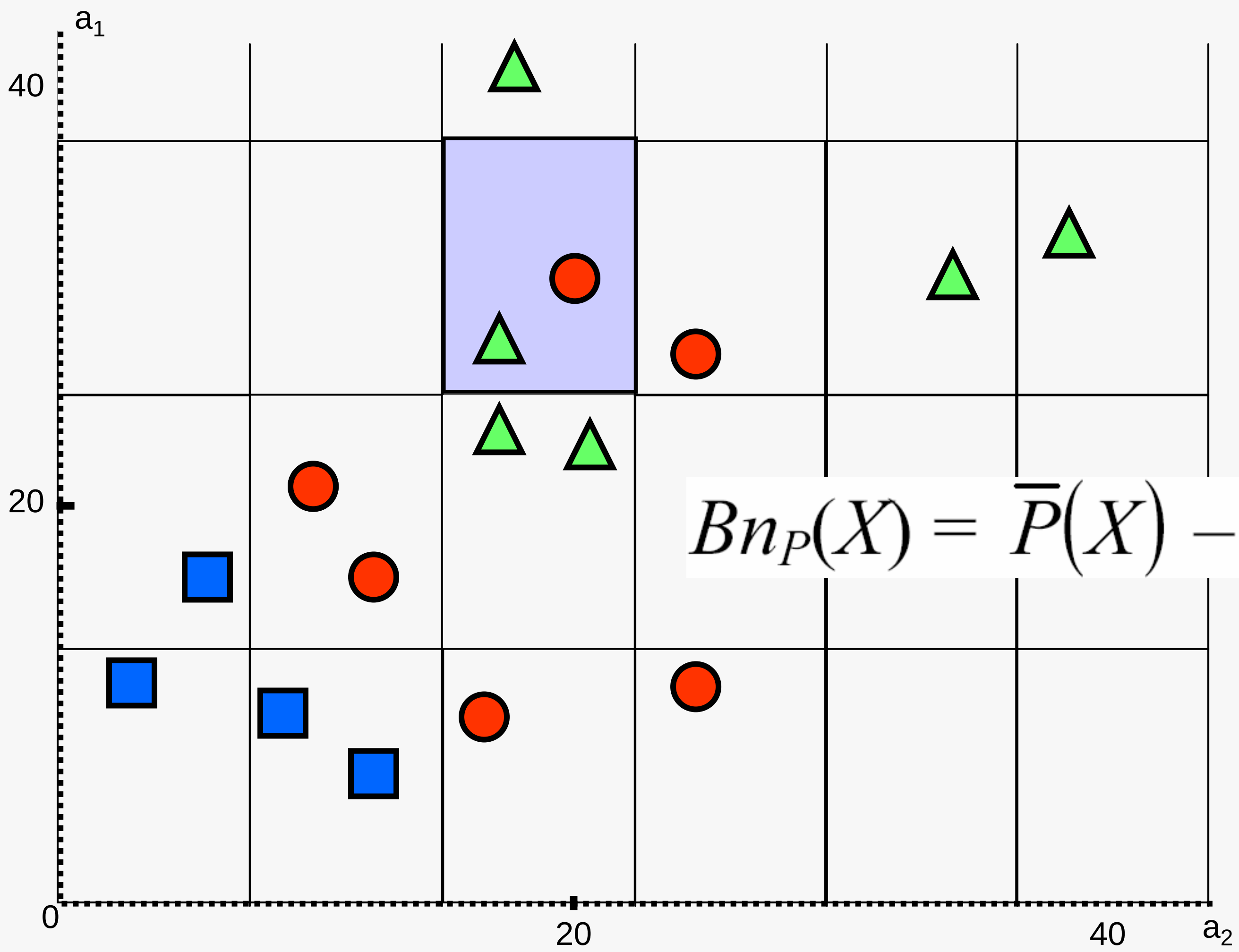
## Upper approximation of class Medium




$$\overline{P}(X) = \bigcup_{x \in X} I_P(x)$$

# CRSA – illustration of formal definitions

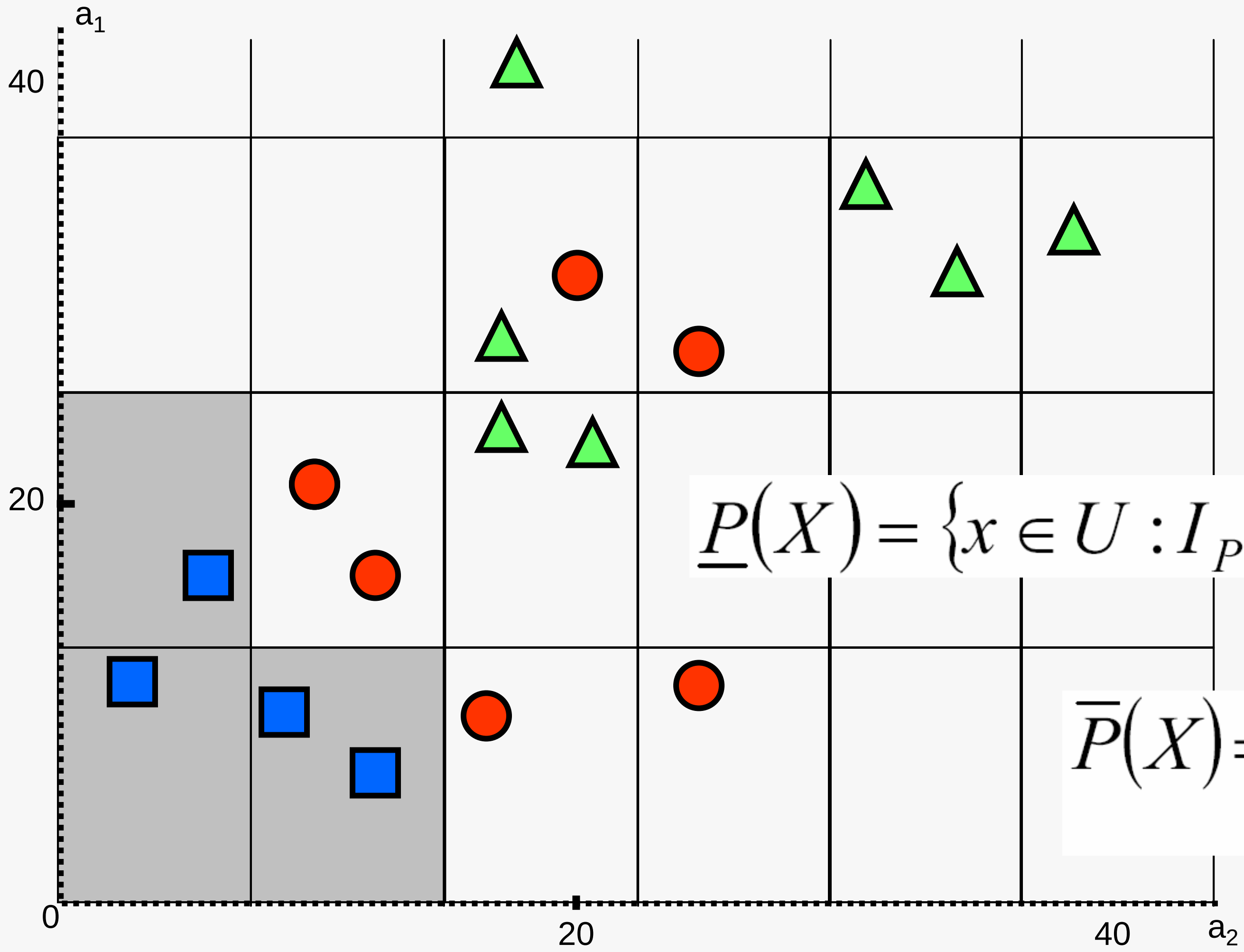
Boundary set of classes High and Medium




$$Bn_P(X) = \overline{P}(X) - \underline{P}(X)$$

# CRSA – illustration of formal definitions

Lower = Upper approximation of class Low




$$\underline{P}(X) = \{x \in U : I_P(x) \subseteq X\}$$

$$\overline{P}(X) = \bigcup_{x \in X} I_P(x)$$

# CRSA – formal definitions

- Inclusion and complementarity properties of rough approximations

$$\underline{P}(X) \subseteq X \subseteq \overline{P}(X) \qquad \underline{P}(X) = U - \overline{P}(U - X)$$

- Accuracy measures
  - Accuracy and quality of approximation of $X \subset U$ by attributes $P \subseteq C$

$$\alpha_P(X) = \frac{\left|\underline{P}(X)\right|}{\left|\overline{P}(X)\right|} \qquad \gamma_P(X) = \frac{\left|\underline{P}(X)\right|}{|X|}$$

  - *Quality of approximation of classification* $\mathbf{Cl}=\{Cl_t,\ t=1,\dots n\}$ by attributes $P \subseteq C$

$$\gamma_P(\mathbf{Cl}) = \frac{\sum_{t=1}^{n}\left|\underline{P}(Cl_t)\right|}{|U|}$$

  - Rough membership of $x \in U$ to $X \subset U$, given $P \subseteq C$

$$\mu_X^P(x) = \frac{\left|X \cap I_P(x)\right|}{\left|I_P(x)\right|}$$

# CRSA – formal definitions

- Monotonicity property with respect to the cardinality of $P \subseteq C$:

  for any $R \subseteq P \subseteq C$ it holds:

$$\underline{R}(X) \subseteq \underline{P}(X), \quad \overline{R}(X) \supseteq \overline{P}(X)$$

- ***Cl**-reduct* of $P \subseteq C$, denoted by $RED_{\mathbf{Cl}}(P)$, is a minimal subset $P'$ of $P$ which keeps the quality of classification ***Cl*** unchanged, i.e.

$$\gamma_{P'}(\mathbf{Cl}) = \gamma_P(\mathbf{Cl})$$

- ***Cl***-core is the intersection of all the ***Cl***-reducts of *P*:

$$CORE_{\mathbf{Cl}}(P) = \bigcap RED_{\mathbf{Cl}}(P)$$

# CRSA – decision rules induced from rough approximations

- Certain decision rule supported by objects from lower approximation of $Cl_t$ (discriminant rule)

$$if \;\; x_{q_1} = r_{q_1} \;\; and \;\; x_{q_2} = r_{q_2} \;\; and \ldots x_{q_p} = r_{q_p}, \;\; then \;\; x \in Cl_t$$

- Possible decision rule supported by objects from upper approximation of $Cl_t$ (partly discriminant rule)

$$if \;\; x_{q_1} = r_{q_1} \;\; and \;\; x_{q_2} = r_{q_2} \;\; and \ldots x_{q_p} = r_{q_p}, \;\; then \;\; x \in Cl_t$$

- Approximate decision rule supported by objects from the boundary of $Cl_t$

$$if \;\; x_{q_1} = r_{q_1} \;\; and \;\; x_{q_2} = r_{q_2} \;\; and \ldots x_{q_p} = r_{q_p}, \;\; then \;\; x \in Cl_t \; or \; Cl_s \; or \ldots Cl_u$$

where $\{q_1, q_2, \ldots, q_p\} \subseteq C, \; (r_{q_1}, r_{q_2}, \ldots, r_{q_p}) \in V_{q_1} \times V_{q_2} \times \ldots \times V_{q_p}$

$Cl_t, Cl_s, \ldots, Cl_u$ are classes to which belong inconsistent objects supporting this rule

- Rule strength – percentage of objects in $U$ supporting the rule

# Measures characterizing decision rules in system $S=\langle U, C, D\rangle$

- *Support* of a rule $\Phi \to \Psi$: $$supp_S(\Phi,\Psi) = card(\|\Phi \wedge \Psi\|_S)$$
- *Strength* of a rule $\Phi \to \Psi$: $$\sigma_S(\Phi,\Psi) = \frac{supp_S(\Phi,\Psi)}{card(U)}$$
- *Certainty (or confidence) factor* of a rule $\Phi \to \Psi$ (Łukasiewicz, 1913): $$cer_S(\Phi,\Psi) = \frac{supp_S(\Phi,\Psi)}{card(\|\Phi\|_S)}$$
- *Coverage factor* of a rule $\Phi \to \Psi$: $$cov_S(\Phi,\Psi) = \frac{supp_S(\Phi,\Psi)}{card(\|\Psi\|_S)}$$
- Relation between certainty, coverage and Bayes theorem:

$$cer_S(\Phi,\Psi) = Pr(\Psi|\Phi) = \frac{Pr(\Psi \wedge \Phi)}{Pr(\Phi)}, \quad cov_S(\Phi,\Psi) = Pr(\Phi|\Psi) = \frac{Pr(\Phi \wedge \Psi)}{Pr(\Psi)}$$

# CRSA – summary of useful results

- Characterization of decision classes (even in case of inconsistency) in terms of chosen attributes by lower and upper approximation
- Measure of the quality of approximation indicating how good the chosen set of attributes is for approximation of the classification
- Reduction of knowledge contained in the table to the description by relevant attributes belonging to reducts
- The core of attributes indicating indispensable attributes
- Decision rules induced from lower and upper approximations of decision classes show classification patterns existing in data

# Algorytm indukcji reguł decyzyjnych z przybliżeń klas decyzyjnych – LEM2

- Definicje i oznaczenia:

  Przykład – obiekt z tablicy $S=\langle U, C, D\rangle$

  $B$ – niepuste dolne lub górne przybliżenie, lub brzeg klasy decyzyjnej $Cl_t$, czyli zbiór przykładów pozytywnych

  $q$ – atrybut ze zbioru $C$

  $V_q$ – dziedzina wartości atrybutu $q \in C$

  $(q=v_q)$ – warunek elementarny części warunkowej reguły, $q \in C$, $v_q \in V_q$

  $t = (q,v_q)$ – para „atrybut-wartość” tworząca warunek elementarny $(q=v_q)$

  $[t]$ – zbiór przykładów spełniających odpowiedni warunek elementarny $(q=v_q)$

  $T$ – zbiór par $t$ „atrybut-wartość” zwany koniunkcją, tworzący część warunkową reguły: $(q_1=v_{q_1}) \wedge (q_2=v_{q_2}) \wedge \ldots \wedge (q_p=v_{q_p})$

  $[T]$ – zbiór przykładów spełniających koniunkcję $(q_1=v_{q_1}) \wedge (q_2=v_{q_2}) \wedge \ldots \wedge (q_p=v_{q_p})$, czyli zbiór przykładów „pokrytych” przez $T$

# Algorytm LEM2

- Zbiór $B$ zależy od koniunkcji $T$ par „atrybut-wartość" wtedy i tylko wtedy, gdy

$$\emptyset \neq [T] = \bigcap_{t \in T} [t] \subseteq B$$

- Koniunkcja $T$ jest minimalna wtedy i tylko wtedy, gdy $B$ zależy od $T$, a usunięcie dowolnej pary $t$ powoduje, że $B$ nie zależy od koniunkcji $T' = T - \{t\}$
- $\boldsymbol{T}$ jest zbiorem koniunkcji $T$ stanowiącym tzw. lokalne pokrycie zbioru $B$ wtedy i tylko wtedy, gdy spełnione są następujące warunki:
  - każda koniunkcja $T \in \boldsymbol{T}$ jest minimalna
  - $\bigcup_{T \in \boldsymbol{T}} [T] = B$
  - $\boldsymbol{T}$ jest zbiorem o najmniejszej liczbie koniunkcji $T$
- Koniunkcja $T \in \boldsymbol{T}$ dla $B = \underline{C}(Cl_t)$ tworzy regułę decyzyjną pewną:

*Jeżeli* $(x_{q_1}=v_{q_1}) \wedge (x_{q_2}=v_{q_2}) \wedge \ldots \wedge (x_{q_p}=v_{q_p})$, *to* $x$ należy do $Cl_t$

- Koniunkcja $T \in \boldsymbol{T}$ dla $B = \overline{C}(Cl_t)$ tworzy regułę decyzyjną możliwą:

*Jeżeli* $(x_{q_1}=v_{q_1}) \wedge (x_{q_2}=v_{q_2}) \wedge \ldots \wedge (x_{q_p}=v_{q_p})$, *to* $x$ być może należy do $Cl_t$

# Algorytm LEM2

- Koniunkcja $T \in \boldsymbol{T}$ dla $B = Bn_C(Cl_t)$ tworzy regułę decyzyjną przybliżoną:

  *Jeżeli* $(x_{q_1}=v_{q_1}) \wedge (x_{q_2}=v_{q_2}) \wedge \ldots \wedge (x_{q_p}=v_{q_p})$, *to x* należy do $Cl_t \cup Cl_s \cup \ldots \cup Cl_u$

- Algorytm LEM2 buduje pojedyncze lokalne pokrycie $\boldsymbol{T}$ dla każdej klasy decyzyjnej, a dokładniej, dla każdego dolnego lub górnego przybliżenia klasy $Cl_t$, lub brzegu klasy $Cl_t$ (zbioru $B$)

- Algorytm LEM2 uruchamia się dla danego zbioru $B$, a w wyniku otrzymuje się minimalny zbiór reguł pokrywających wszystkie obiekty z tego zbioru

# Algorytm LEM2

**Procedure** LEM2

(**input:** zbiór $B$,

**output:** pojedyncze lokalne pokrycie $\boldsymbol{T}$ dla zbioru $B$);

**begin**

$G := B$; {zbiór przykładów nie pokrytych dotychczas przez koniunkcje $T \in \boldsymbol{T}$}

$\boldsymbol{T} := \varnothing$; {zbiór koniunkcji $T$ tworzący aktualne pokrycie zbioru $B-G$}

**while** $G \neq \varnothing$

**begin**

$T := \varnothing$; {koniunkcja warunków elementarnych będąca kandydatem na część warunkową reguły}

$T(G) := \{t : [t] \cap G \neq \varnothing\}$; {zbiór potencjalnych warunków elementarnych dla nie pokrytych przykładów}

**while** $T = \varnothing$ **or** $[T] \not\subseteq B$ **do**

# Algorytm LEM2

**while** $T = \emptyset$ **or** $[T] \not\subseteq B$ **do**

**begin**

wybierz warunek $t \in T(G)$ taki, że wyrażenie $|[T \cup \{t\}] \cap G|$ ma wartość maksymalną; jeśli więcej niż jeden z warunków maksymalizuje powyższe wyrażenie, to wybierz ten, dla którego $|[T \cup \{t\}]|$ ma wartość minimalną; w przypadku niejednoznaczności wybierz pierwszy z rozważanych warunków;

$T := T \cup \{t\}$; {dołącz najlepszy warunek $t$ do koniunkcji $T$}

$G := G \cap [t]$; {ogranicz zbiór przykładów dostarczających nowych warunków elementarnych}

$T(G) := \{w : [w] \cap G \neq \emptyset \}$; {uaktualnij listę potencjalnych warunków elementarnych}

$T(G) := T(G) - T$; {usuń z listy warunki elementarne już wybrane do tworzonej koniunkcji}

**end** {while $[T] \not\subseteq B$}

# Algorytm LEM2

**end** {while $[T] \not\subseteq B$}

**for** każdy warunek elementarny $t \in T$ **do**

**if** $[T - \{t\}] \subseteq B$ **then** $T := T - \{t\}$; {usuń nadmiarowe warunki}

$\boldsymbol{T} := \boldsymbol{T} \cup \{T\}$;

$G := B - \bigcup_{T \in \boldsymbol{T}} [T]$;

**end** {while $G \neq \emptyset$}

**for** każda koniunkcja $T \in \boldsymbol{T}$ **do**

**if** $\bigcup_{K \in \boldsymbol{T} - \{T\}} [K] = B$ **then** $\boldsymbol{T} := \boldsymbol{T} - \{T\}$; {usuń nadmiarowe reguły}

utwórz zbiór reguł $R$ na podstawie wszystkich koniunkcji $T \in \boldsymbol{T}$;

**end** {procedure}

# Przykład

| Przykłady (obiekty) | Atrybuty warunkowe | | | | Decyzja |
|---|---|---|---|---|---|
| | Temperatura | Hemoglobina | Ciśnienie | Dotlenienie | Samopoczucie |
| a | niska | dobra | niskie | dobre | Słabe |
| b | niska | dobra | normalne | złe | Słabe |
| c | normalna | b.dobra | niskie | b.dobre | Słabe |
| d | normalna | b.dobra | niskie | b.dobre | Dobre |
| e | niska | b.dobra | normalne | b.dobre | Dobre |
| f | niska | b.dobra | normalne | dobre | Dobre |
| g | normalna | dobra | normalne | b.dobre | Dobre |
| h | normalna | niska | wysokie | b.dobre | Złe |
| i | wysoka | b.dobra | wysokie | dobre | Złe |

# Przykład

- *C*-dolne przybliżenie klasy *Słabe samopoczucie*: $B = \{a, b\}$
- $T(G)$ = {(Temperatura=niska), (Hemoglobina=dobra), (Ciśnienie=niskie), (Ciśnienie=normalne), (Dotlenienie=dobre), (Dotlenienie=złe)}
- $B = G = \{a, b\}$
- Pokrycia warunków elementarnych: $[T \cup \{t\}]$, $|[T \cup \{t\}] \cap G|$, $|[T \cup \{t\}]|$

$T(G)$:

| | | | |
|---|---|---|---|
| [(Temperatura=niska)] | = {a, b, e, f}, | **2**, | 4 |
| [(Hemoglobina=dobra)] | = {a, b, g}, | **2**, | **3** |
| [(Ciśnienie=niskie)] | = {a, c, d}, | 1, | 3 |
| [(Ciśnienie=normalne)] | = {b, e, f, g}, | 1, | 4 |
| [(Dotlenienie=dobre)] | = {a, f, i}, | 1, | 3 |
| [(Dotlenienie=złe)] | = {b}, | 1, | 1 |

- Procedura wybiera warunki (Temperatura=niska), (Hemoglobina=dobra) ponieważ ich pokrycia mają maksymalny przekrój z *G*
- Z powodu remisu stosuje się kryterium minimalnej liczności pokrycia, więc wybrany zostaje (Hemoglobina=dobra)

# Przykład

- $T$ = {(Hemoglobina=dobra)}
- Nadal $G$ = {a, b}, ponieważ $G \cap$ [(Hemoglobina=dobra)] = $G$
- $T(G)$ = $T(G)$ – $T$ = {(Temperatura=niska), (Ciśnienie=niskie), (Ciśnienie=normalne), (Dotlenienie=dobre), (Dotlenienie=złe)}
- Ponieważ [(Hemoglobina=dobra)] $\not\subseteq B$, potrzebna jest następna iteracja
- Tym razem wybrany zostaje warunek elementarny (Temperatura=niska), bo pokrycie tego warunku ma maksymalny przekrój z $G$
- $T$ = {(Hemoglobina=dobra), (Temperatura=niska)}
- $\boldsymbol{T} = T$
- Ponieważ $[T] = [B]$, zatem pojedyncze lokalne pokrycie $B$ dokonane jest za pomocą jednej reguły:

*Jeżeli* (Hemoglobina=dobra) $\wedge$ (Temperatura=niska), to *Słabe samopoczucie*

## Przykład

- Zbiór reguł **pewnych** wygenerowanych przez LEM2:

*Jeżeli* (Hemoglobina=dobra) ∧ (Temperatura=niska), to *Słabe samopoczucie*

{a,b}, *cert*=1, *cov*=2/3, *strength*=2/9

*Jeżeli* (Hemoglobina=b.dobra) ∧ (Ciśnienie=normalne), to *Dobre samopoczucie*

{e,f}, *cert*=1, *cov*=1/2, *strength*=2/9

*Jeżeli* (Temperatura=normalna) ∧ (Hemoglobina=dobra), to *Dobre samopoczucie*

{g}, *cert*=1, *cov*=1/4, *strength*=1/9

*Jeżeli* (Ciśnienie=wysokie), to *Złe samopoczucie*

{h,i}, *cert*=1, *cov*=1, *strength*=2/9

# Przykład

- Zbiór reguł **możliwych** wygenerowanych przez LEM2:

*Jeżeli* (Ciśnienie=niskie), to *Słabe samopoczucie*

{a,c,d}, *cert*=2/3, *cov*=2/3, *strength*=3/9

*Jeżeli* (Dotlenienie=złe), to *Słabe samopoczucie*

{b}, *cert*=1, *cov*=1/3, *strength*=1/9

*Jeżeli* (Dotlenienie=b.dobre) ∧ (Hemoglobina=b.dobra), to *Dobre samopoczucie*

{c,d,e}, *cert*=2/3, *cov*=1/2, strength=3/9

*Jeżeli* (Dotlenienie=dobre) ∧ (Ciśnienie=normalne), to *Dobre samopoczucie*

{f}, *cert*=1, *cov*=1/4, *strength*=1/9

*Jeżeli* (Temperatura=normalna) ∧ (Hemoglobina=dobra), to *Dobre samopoczucie*

{g}, *cert*=1, *cov*=1/4, *strength*=1/9

*Jeżeli* (Ciśnienie=wysokie), to *Złe samopoczucie*

{h,i}, *cert*=1, *cov*=1, *strength*=2/9