# Vytěžování dat

Filip Železný

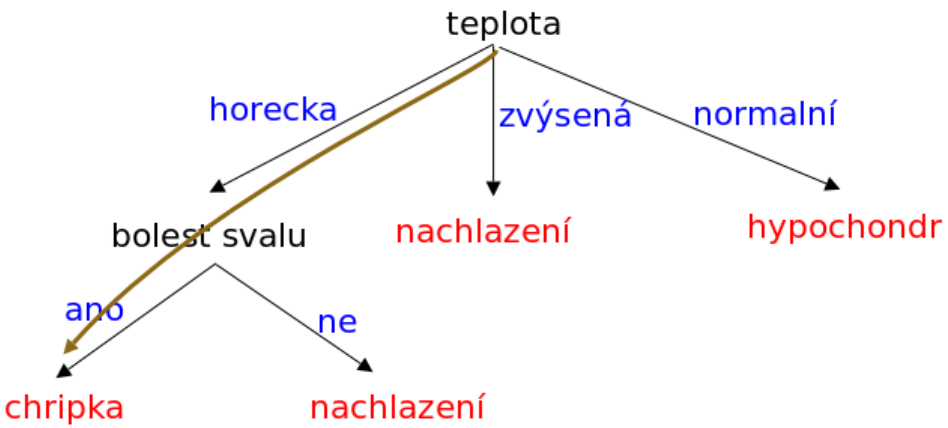
Katedra kybernetiky
skupina Inteligentní Datové Analýzy (IDA)


13. října 2009

# Rozhodovací pravidla

- Strom lze převést na seznam pravidel ve tvaru

**if** ⟨*podmínky*⟩ **then** ⟨*třída*⟩

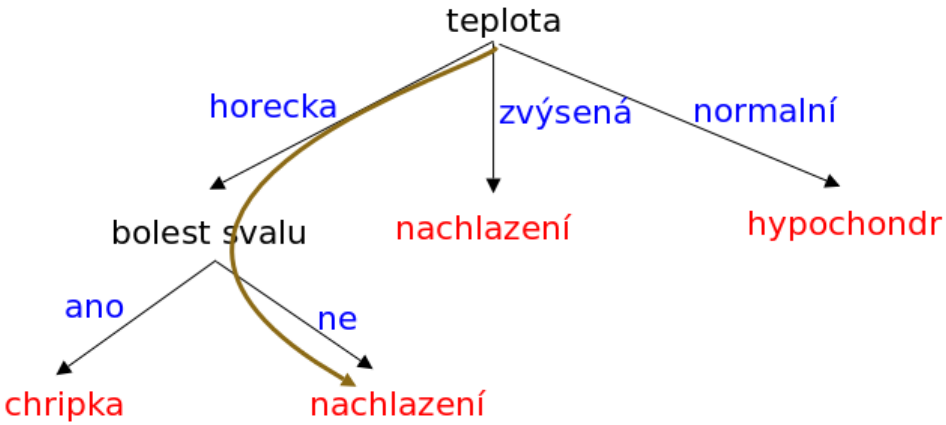



**if** teplota=horečka & bolest_svalů = ano **then** chřipka

# Rozhodovací pravidla

- Strom lze převést na seznam pravidel ve tvaru

**if** ⟨*podmínky*⟩ **then** ⟨*třída*⟩

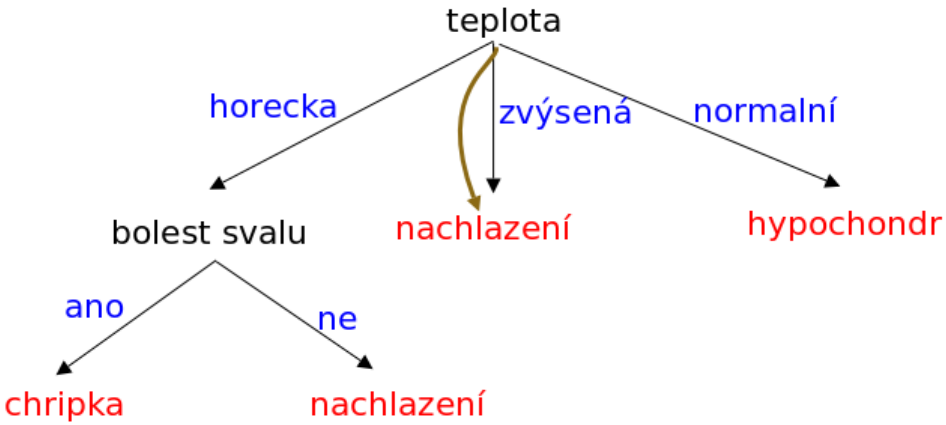



**if** teplota=horečka & bolest_svalů = ano
**then** chřipka
**if** teplota=horečka & bolest_svalů = ne
**then** nachlazení

# Rozhodovací pravidla

- Strom lze převést na seznam pravidel ve tvaru

**if** ⟨*podmínky*⟩ **then** ⟨*třída*⟩

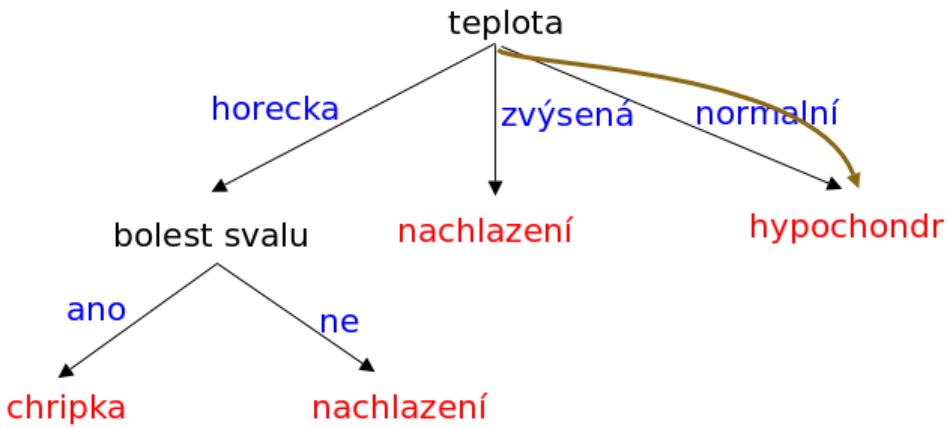



**if** teplota=horečka & bolest_svalů = ano **then** chřipka
**if** teplota=horečka & bolest_svalů = ne **then** nachlazení
**if** teplota=zvýšená **then** nachlazení

# Rozhodovací pravidla

- Strom lze převést na seznam pravidel ve tvaru

**if** ⟨*podmínky*⟩ **then** ⟨*třída*⟩




**if** teplota=horečka & bolest_svalů = ano
**then** chřipka
**if** teplota=horečka & bolest_svalů = ne
**then** nachlazení
**if** teplota=zvýšená **then** nachlazení
**if** teplota=normální **then** hypochondr

- Pravidla lze ale hledat i přímo z dat.

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** příjem = vysoký & bydliště = Praha & pohlaví = M **then** splácí

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** příjem = vysoký & ~~bydliště = Praha~~ & pohlaví = M **then** splácí

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** příjem = vysoký & pohlaví = M **then** splácí

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** příjem = vysoký & ~~pohlaví = M~~ **then** splácí

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** příjem = vysoký **then** splácí

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** ~~příjem = vysoký~~ **then** splácí

# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** ~~příjem = vysoký~~ **then** splácí

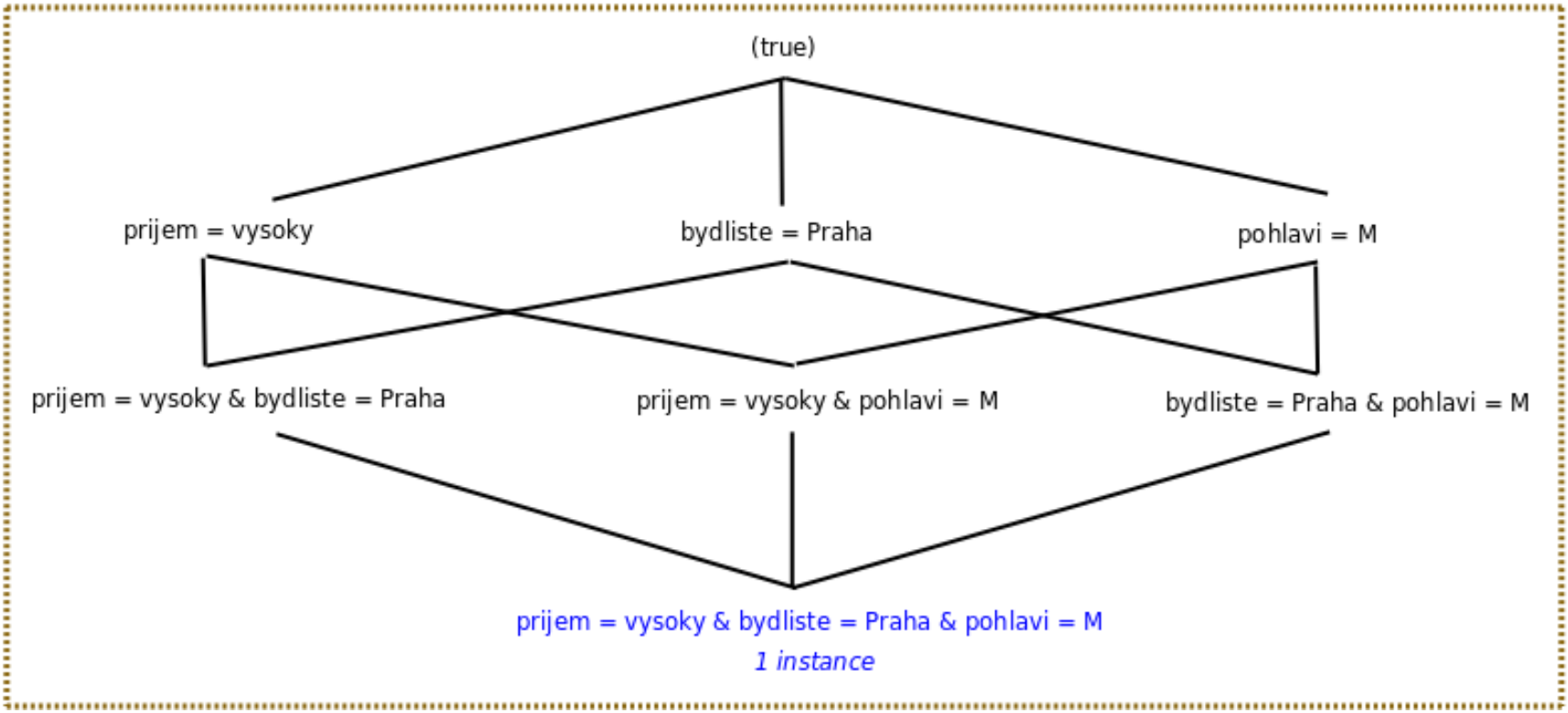
# Hledání pravidla

Chceme najít nejlepší pravidlo pro třídu **splácí**

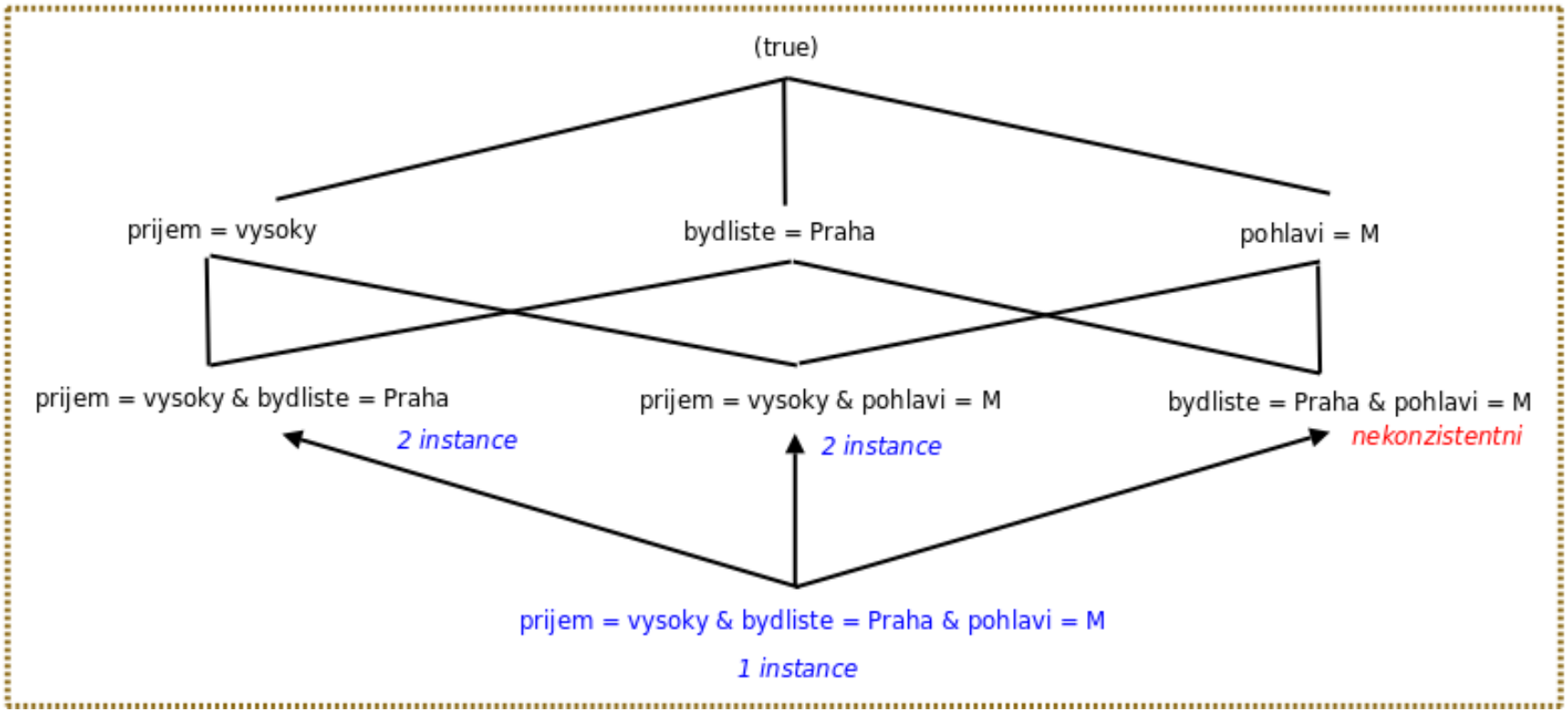
| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

**if** příjem = vysoký **then** splácí

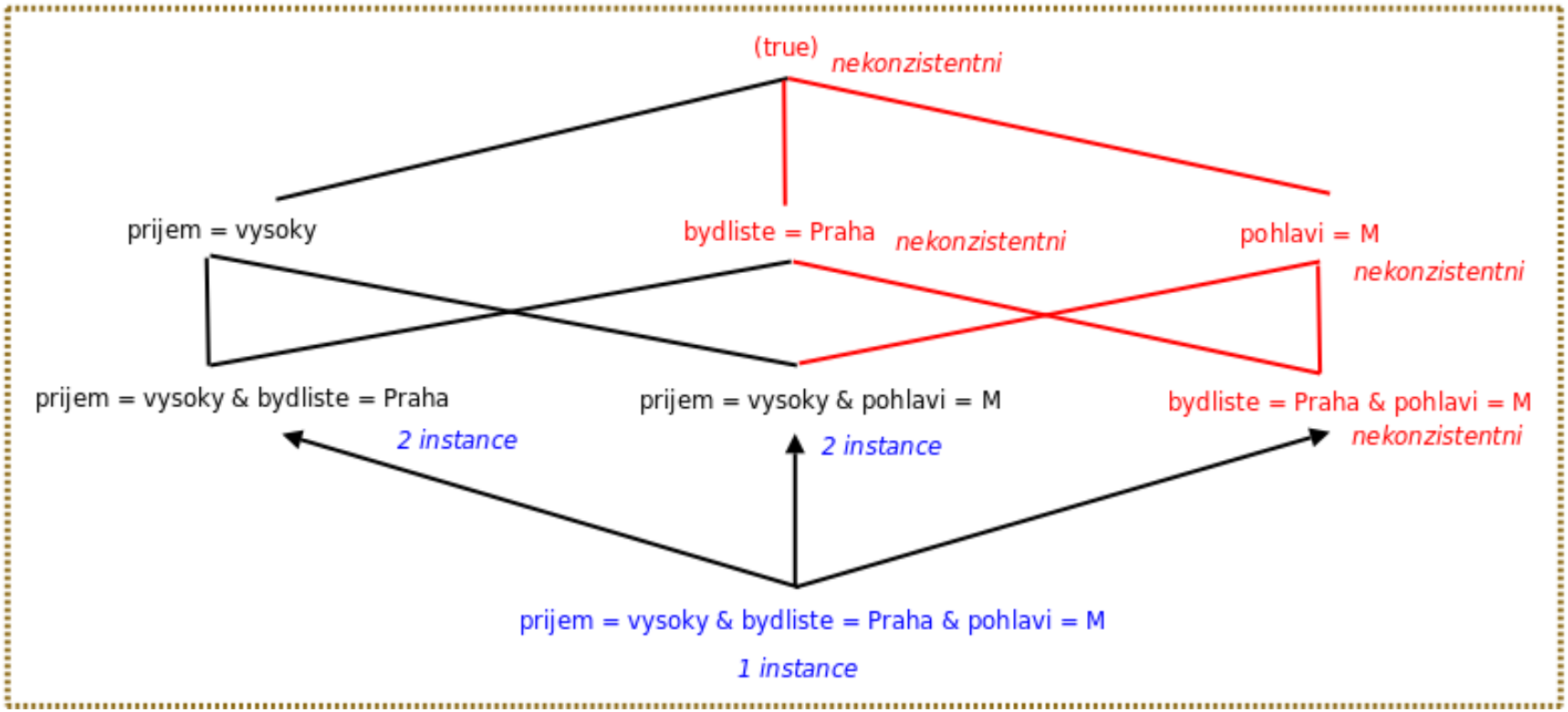
# Zobecňování podmínek



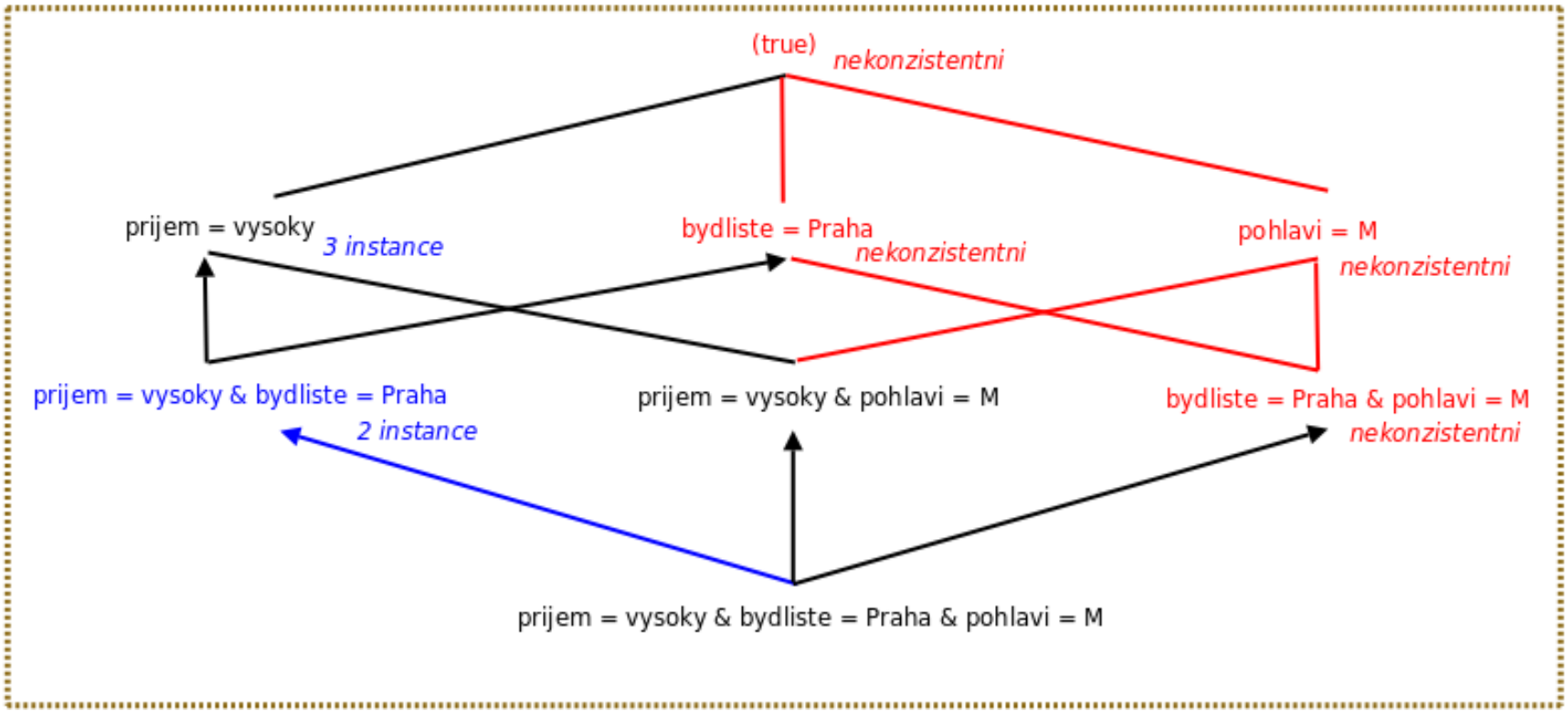
# Zobecňování podmínek



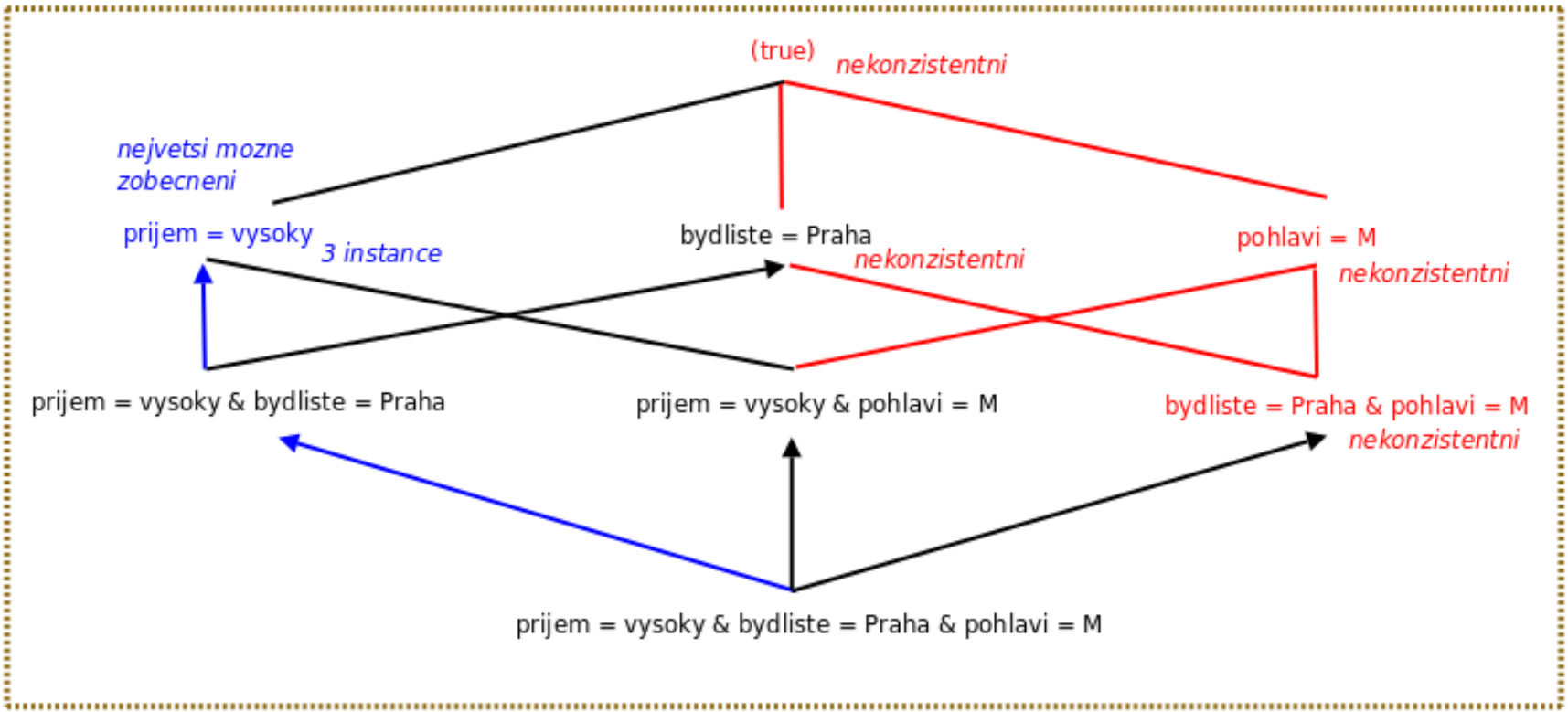
# Zobecňování podmínek



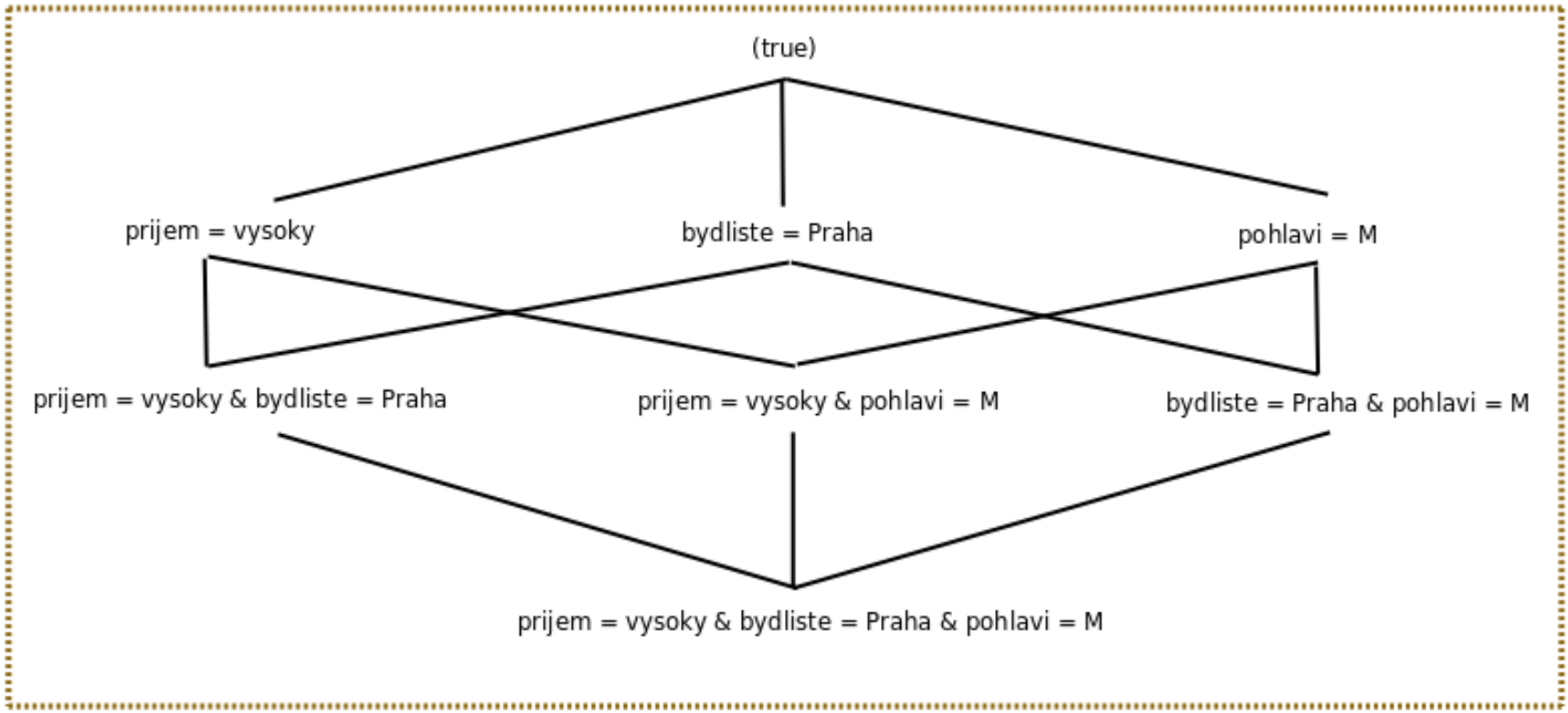
# Zobecňování podmínek



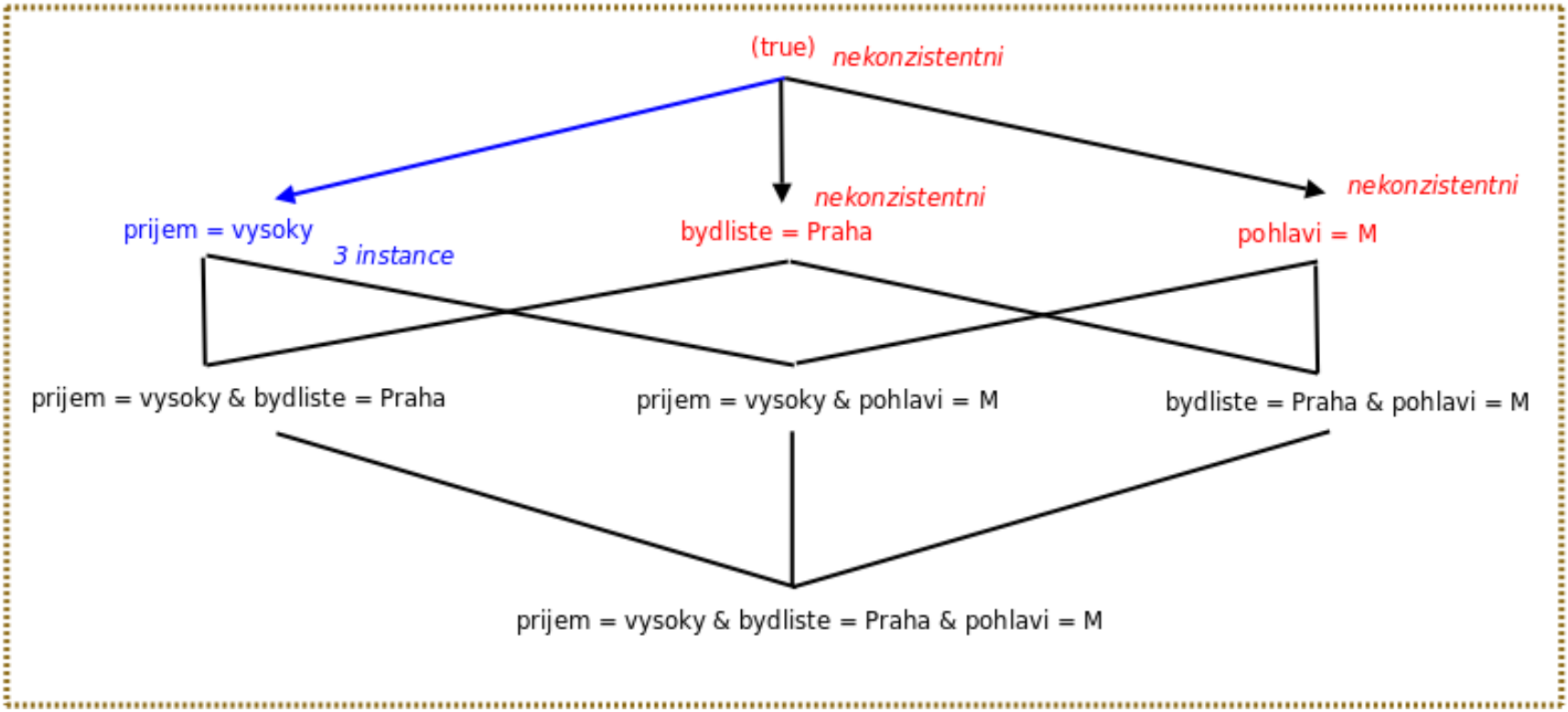
# Zobecňování podmínek



# Alternativa: specializace podmínek

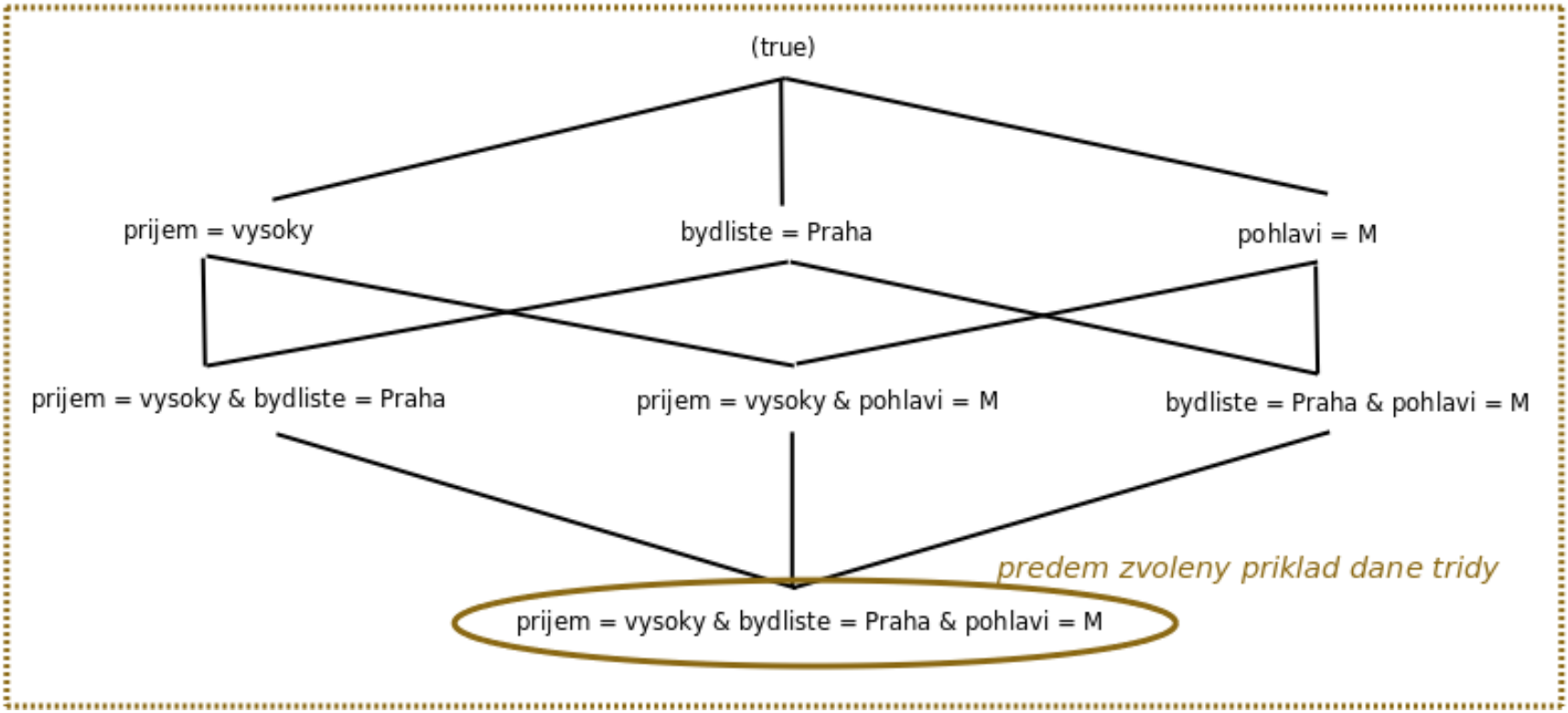


# Alternativa: specializace podmínek

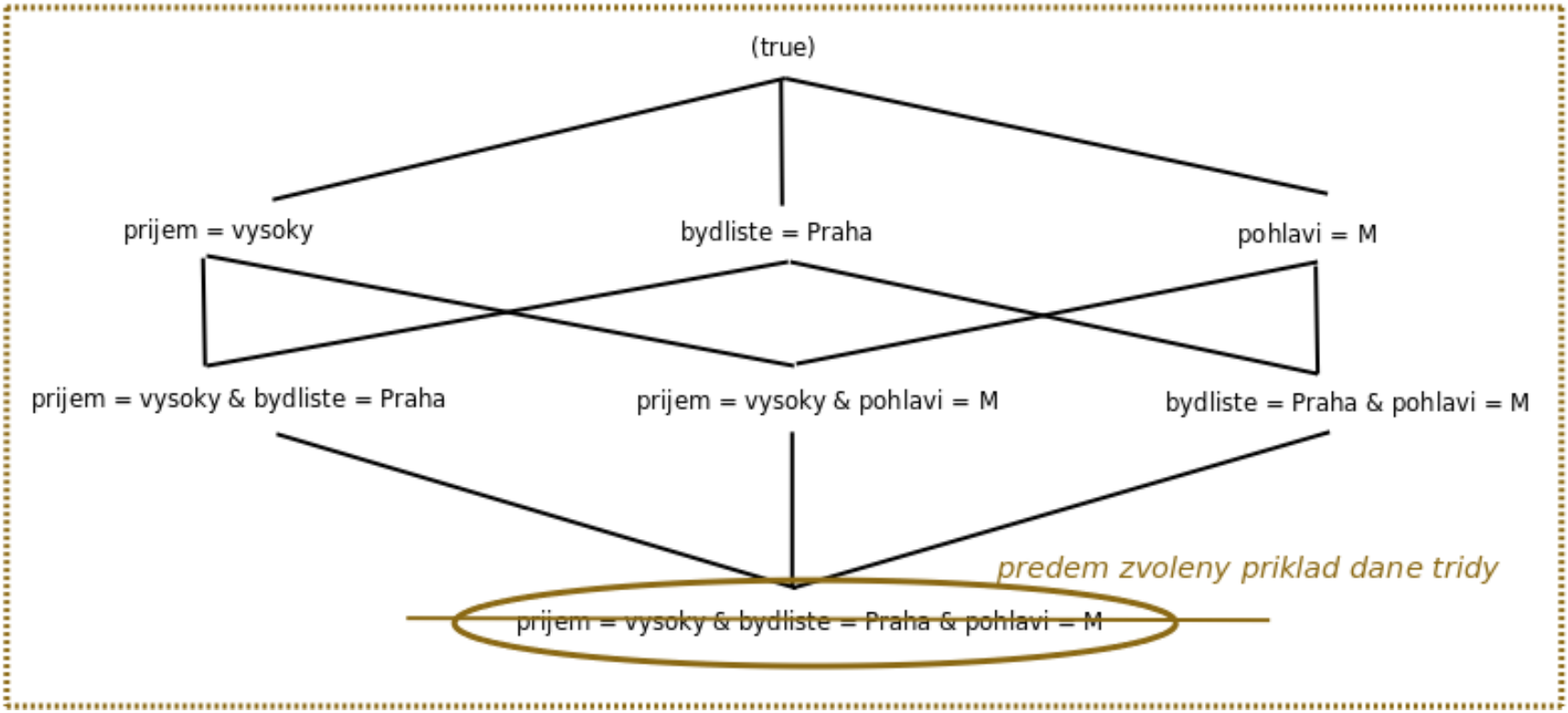



Nemá smysl dále specializovat.

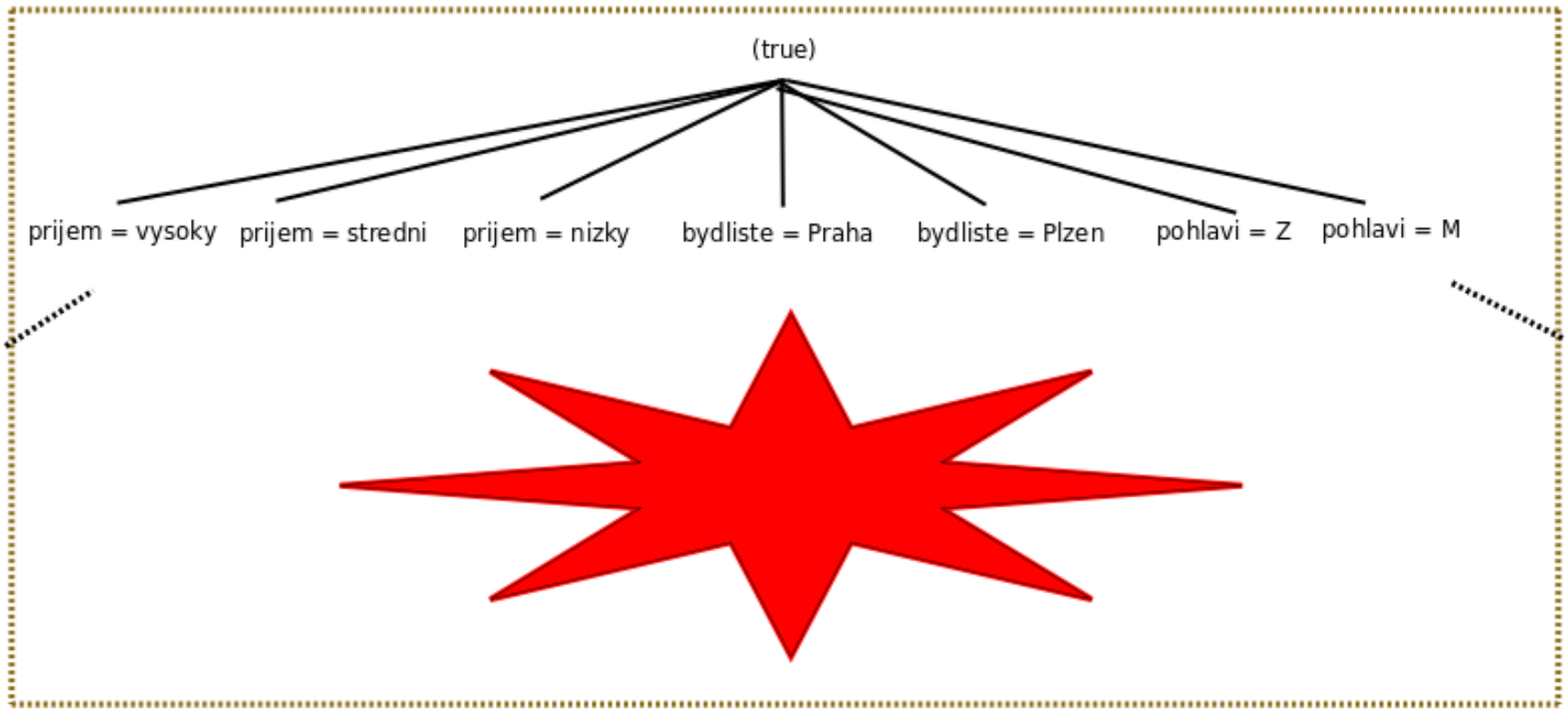
# Specializace bez předem zvoleného příkladu



# Specializace bez předem zvoleného příkladu



# Specializace bez předem zvoleného příkladu




Mnohem větší prohledávací prostor, ale pravidlo může být nakonec lepší.

# Pokrývací strategie

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

# Pokrývací strategie

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

1. **if** příjem = vysoký **then** splácí

# Pokrývací strategie

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

1. **if** příjem = vysoký
   **then** splácí

# Pokrývací strategie

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

1. **if** příjem = vysoký **then** splácí
2. **if** příjem = střední **then** splácí

# Pokrývací strategie

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

1. **if** příjem = vysoký **then** splácí
2. **if** příjem = střední **then** splácí

# Pokrývací strategie

| příjem | bydliště | pohlaví | úvěr |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

1. **if** příjem = vysoký **then** splácí
2. **if** příjem = střední **then** splácí

## Pokrývací strategie

1. Vygeneruj co nejobecnější konzistentní pravidlo a vymaž pokryté příklady.
2. Opakuj krok 1, dokud jsou některé příklady nepokryté.

Tyto kroky postupně uplatni na každou třídu.

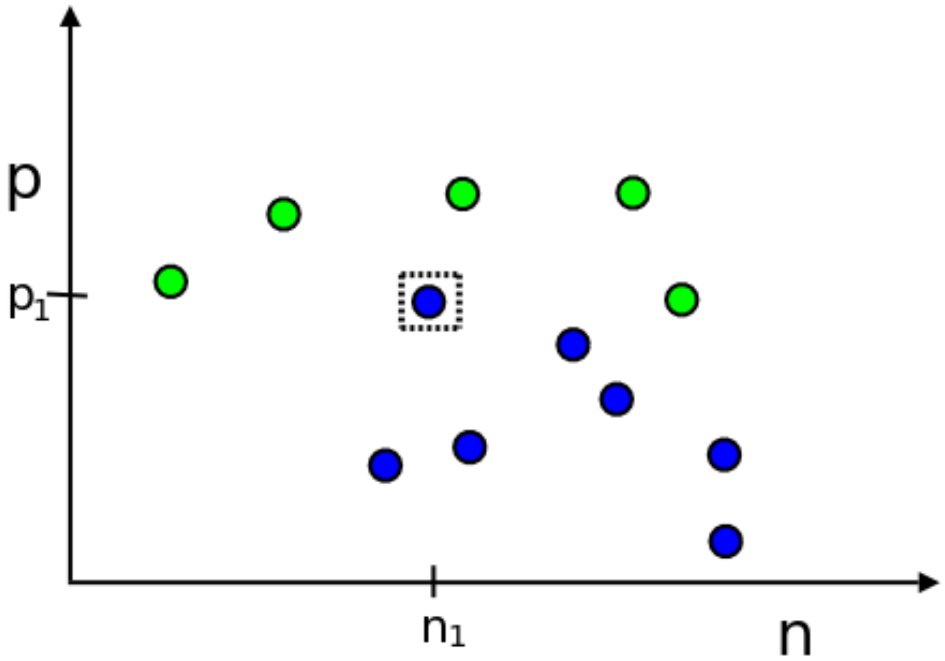
# Algoritmus pro tvorbu seznamu rozhodovacích pravidel

## AQ(D) /* by Richard Michalski */

**Input**: $D$ trénovací data
SeznamPravidel $= \emptyset$
**for** *každou třídu* $t$ **do**
  $P =$ instance třídy $t$ z $D$
  $N =$ ostatní instance $D$
  **repeat**
    Najdi pravidlo $R$ pokrývající nějaké instance z $P$ a nepokrývající žádný příklad z $N$ (konzistentní)
    Přidej $R$ do SeznamPravidel
    Vymaž z $P$ všechny instance pokryté pravidlem $R$
  **until** $P = \emptyset$ ;
**end**

# Separace seznamem pravidel

Separace v prostoru dvou reálných příznaků

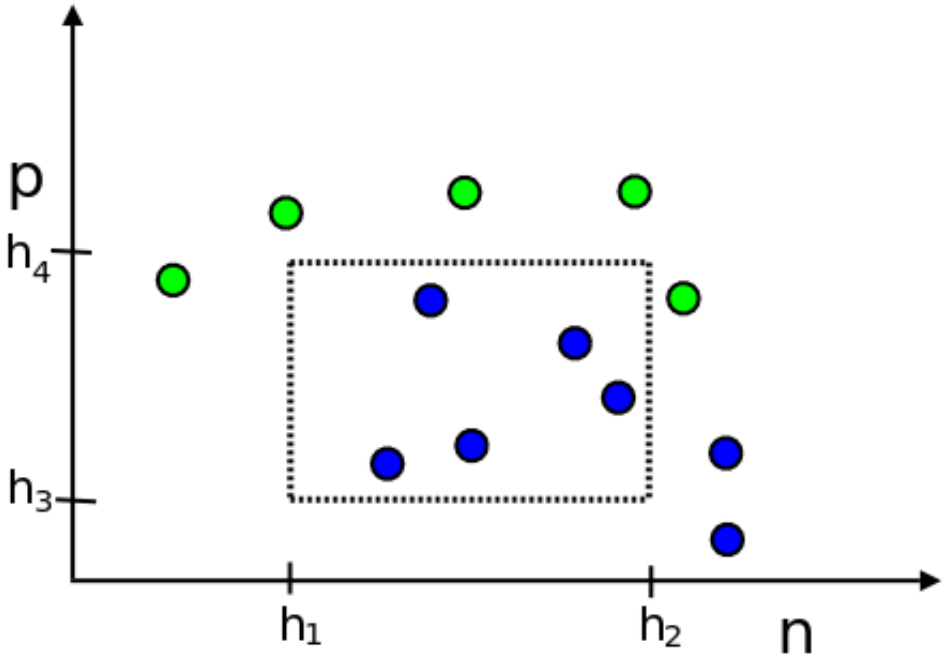



IF $p=p_1$ & $n=n_1$ THEN

Zvolen jeden modrý příklad

# Separace seznamem pravidel

Separace v prostoru dvou reálných příznaků

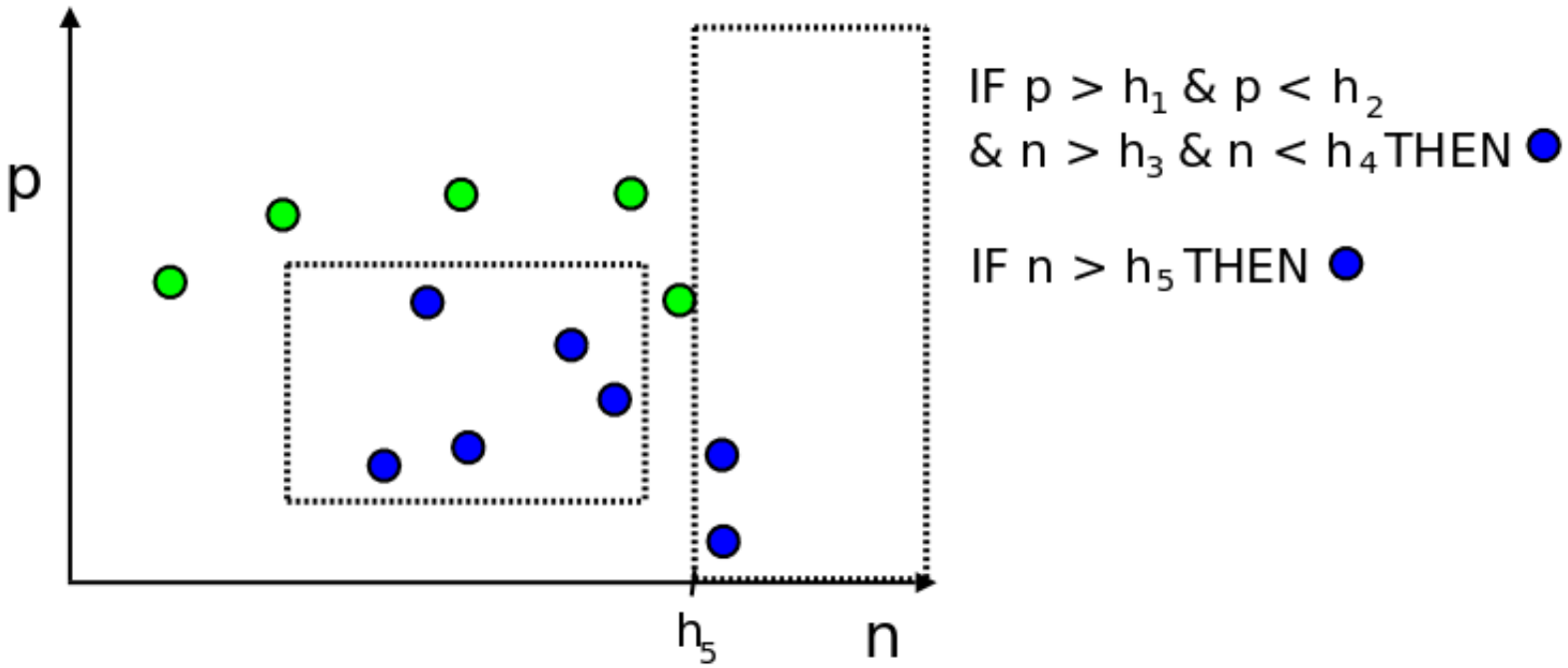



IF $p > h_1$ & $p < h_2$
& $n > h_3$ & $n < h_4$ THEN ●

Zobecnění s použitím hraničních hodnot (předchozí diskretizace)

# Separace seznamem pravidel

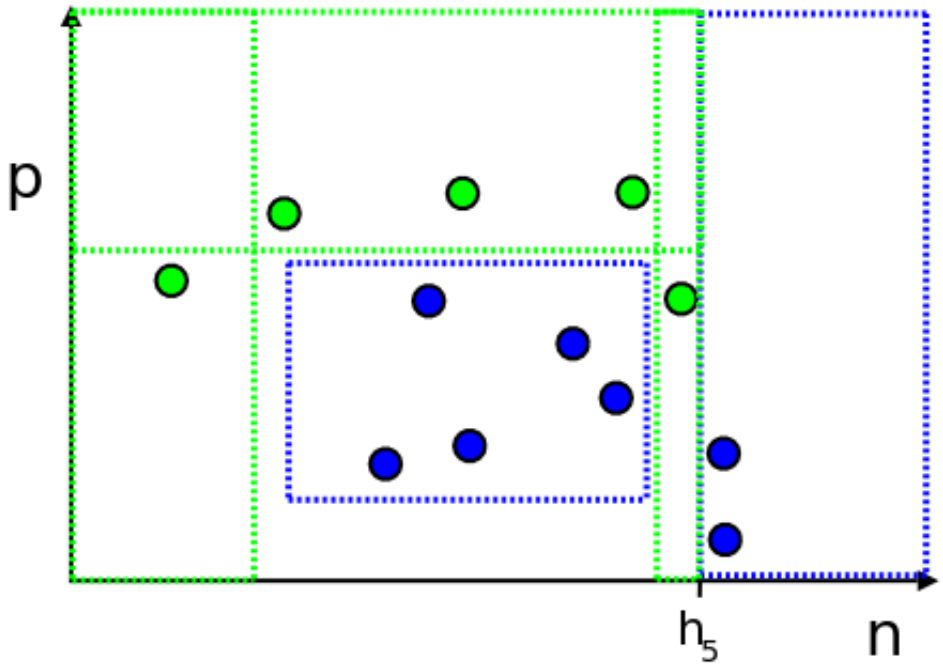
Separace v prostoru dvou reálných příznaků




Přidání dalšího pravidla

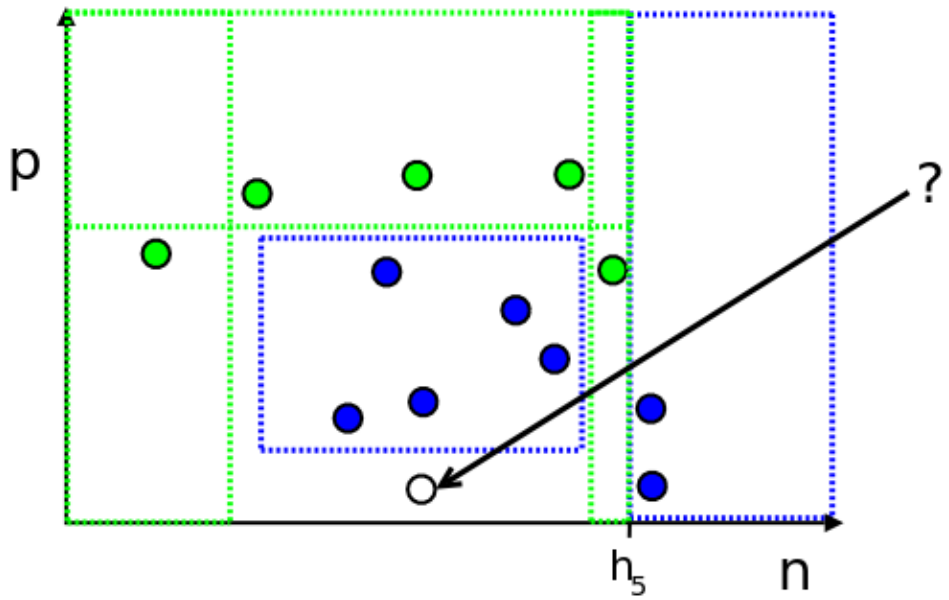
# Separace seznamem pravidel

Separace v prostoru dvou reálných příznaků



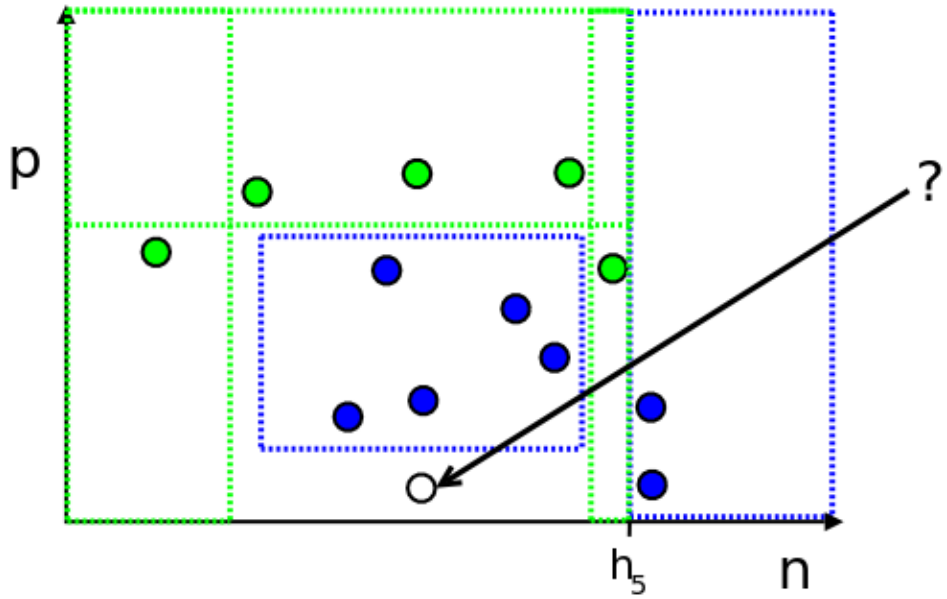

Výsledek pokrývacího algoritmu

# Neúplnost klasifikace podle pravidel




- Některé instance seznam pravidel nemusí rozhodnout!
- Narozdíl od rozhodovacího stromu

# Neúplnost klasifikace podle pravidel

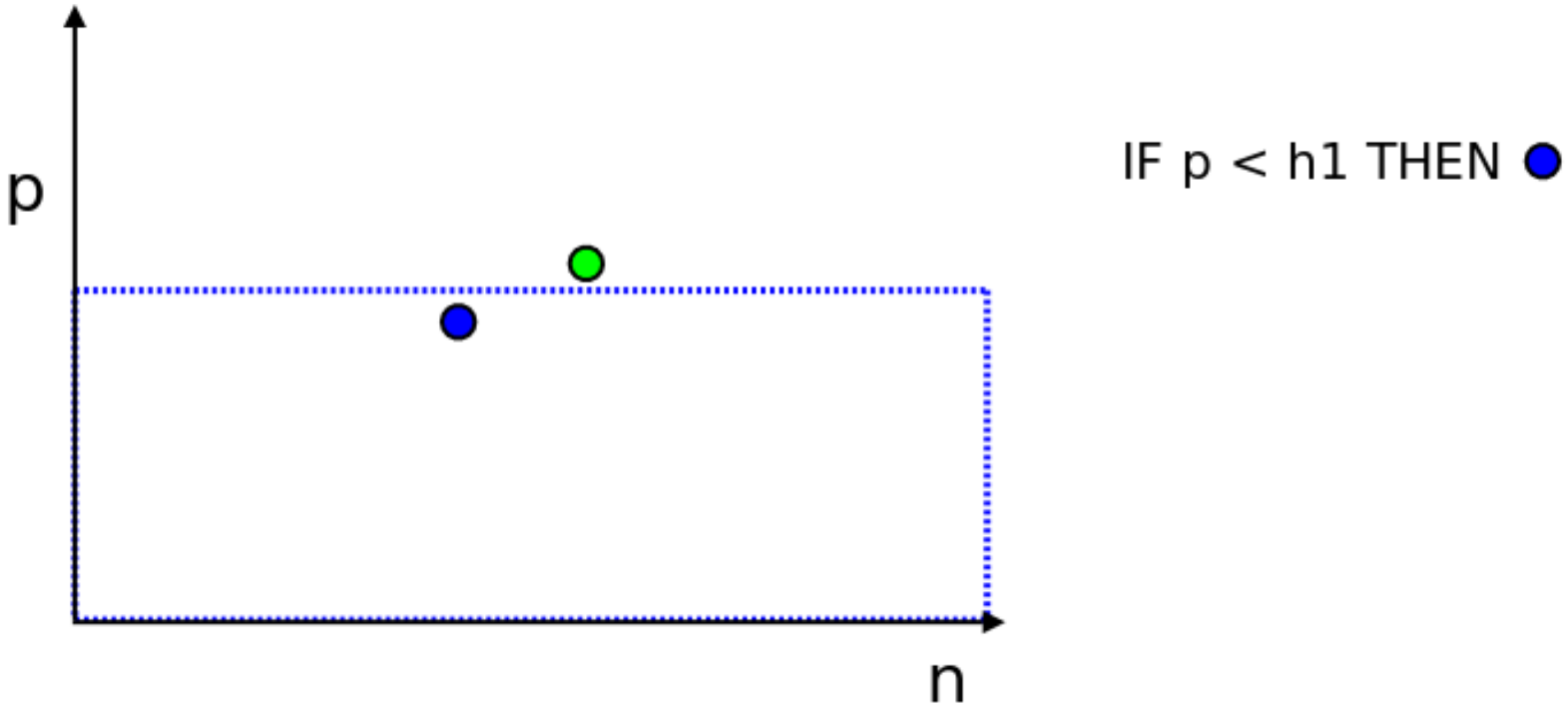



- Řeší se zavedením *implicitního* (default) pravidla

  **if** *true* **then** *většinová třída*

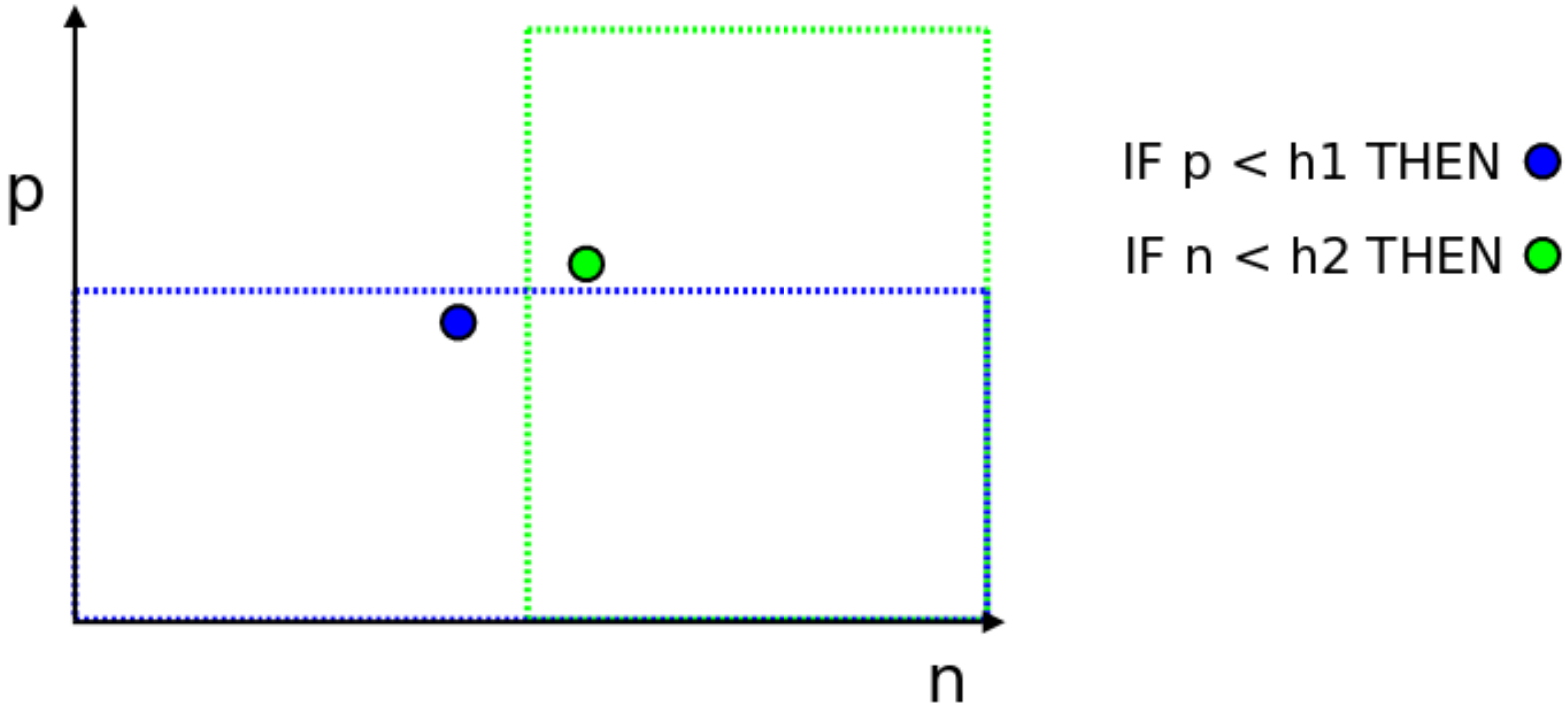
- které se použije, pokud podmínky žádného jiného pravidla neplatí.

# Konflikt pravidel



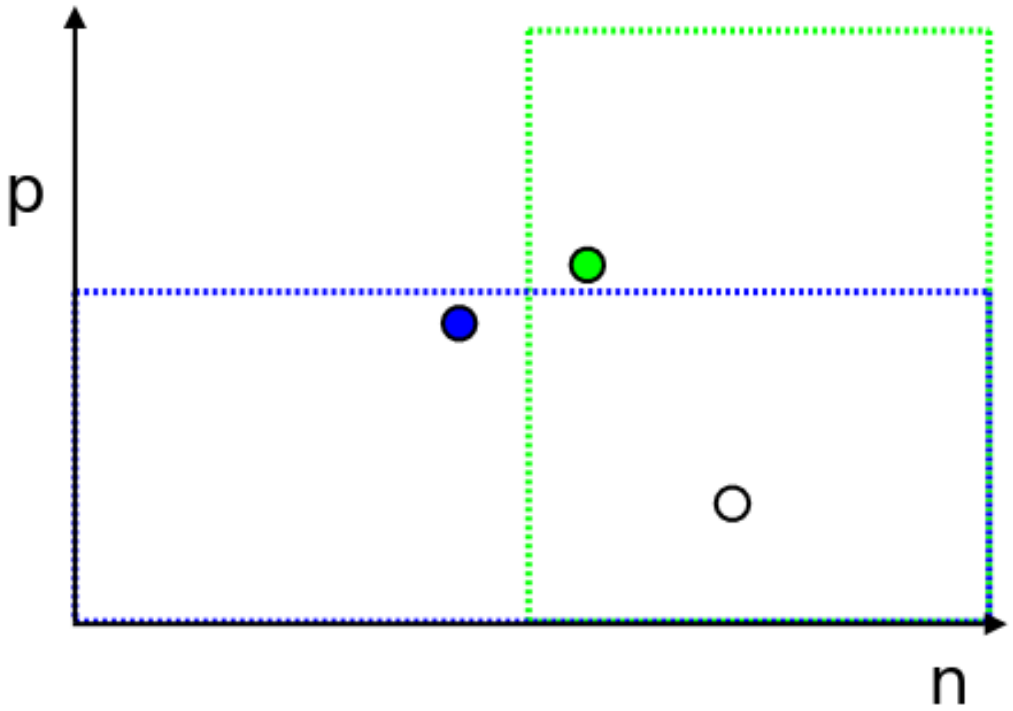

Pravidlo pro modrou třídu

# Konflikt pravidel




Potom pravidlo pro zelenou třídu. Překrytí - konflikt!

# Konflikt pravidel

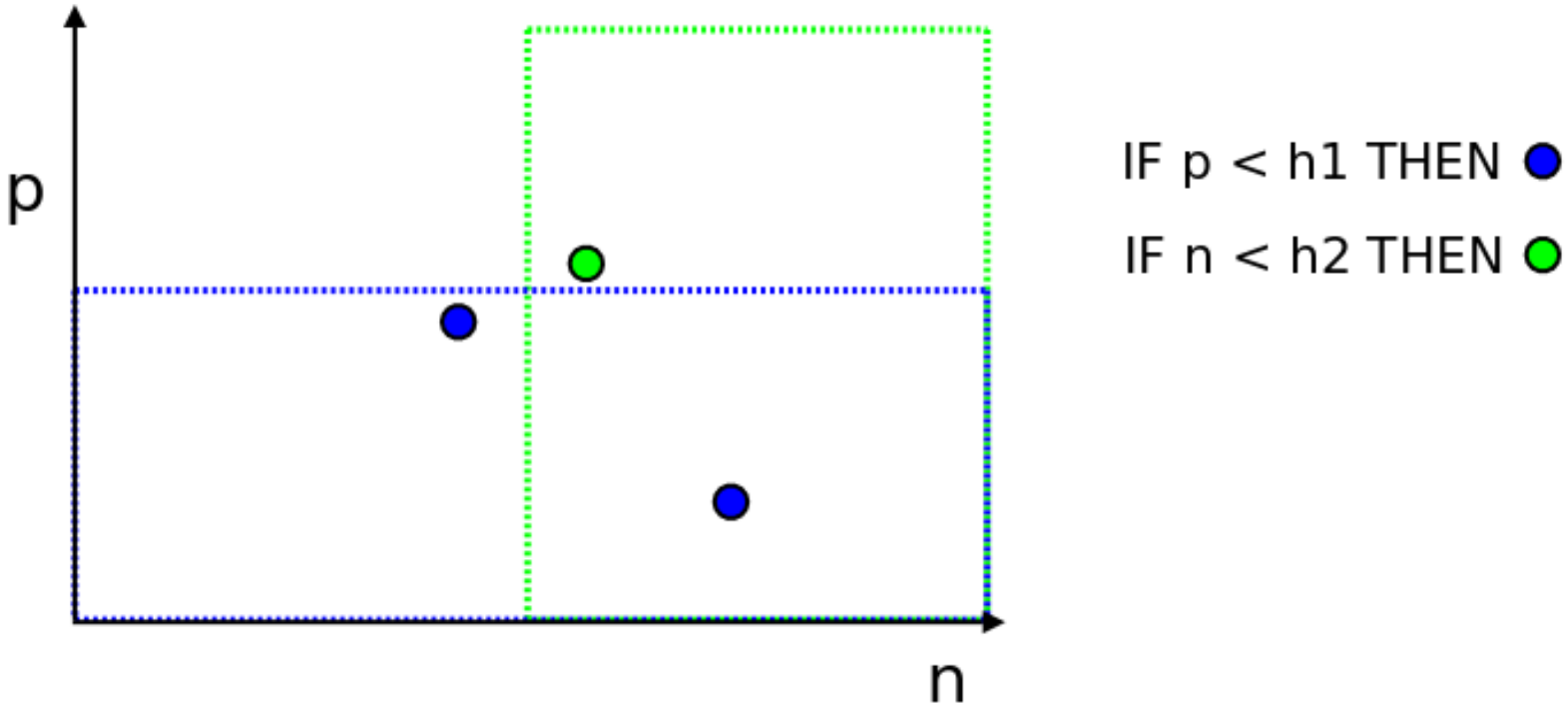



IF p < h1 THEN ●

IF n < h2 THEN ●

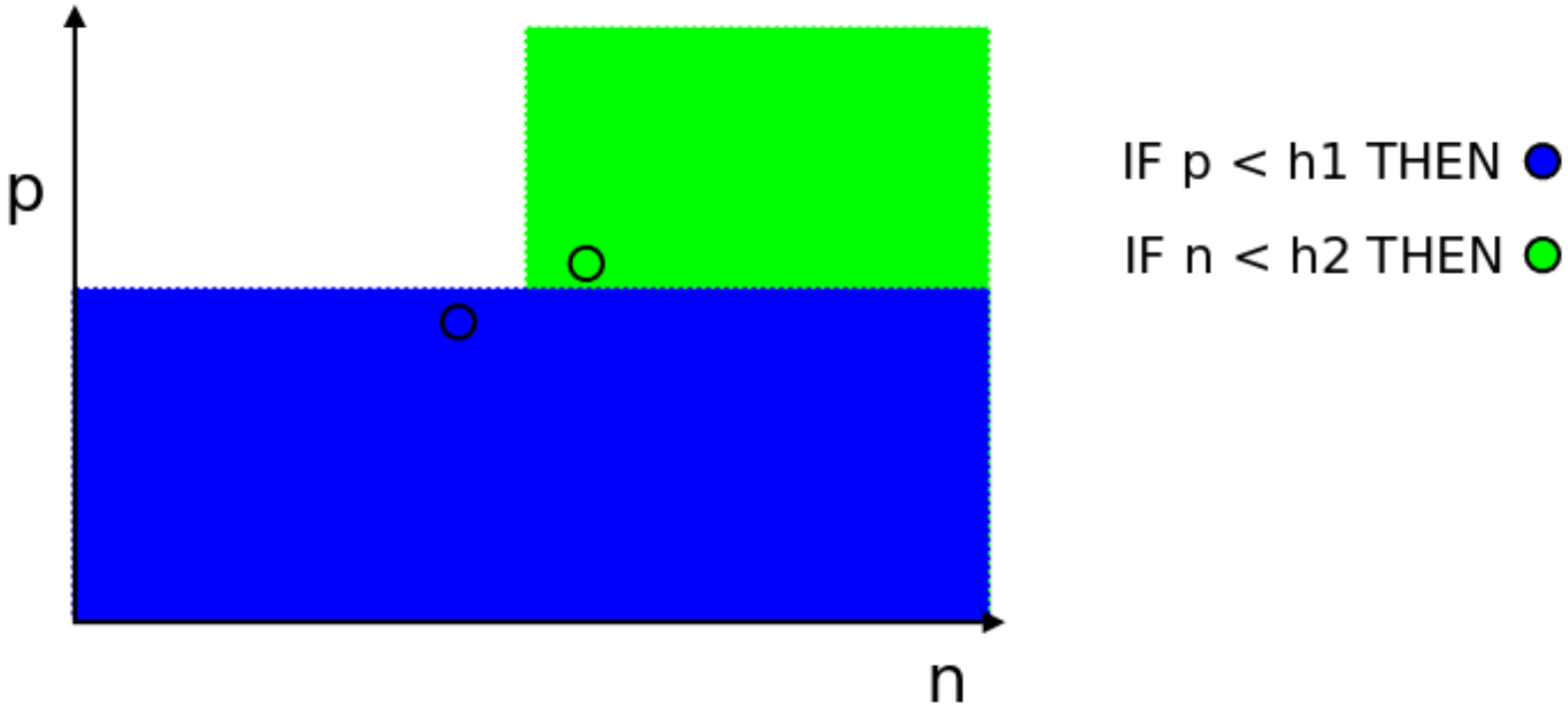
Jak seznamem klasifikovat novou instanci?

# Konflikt pravidel



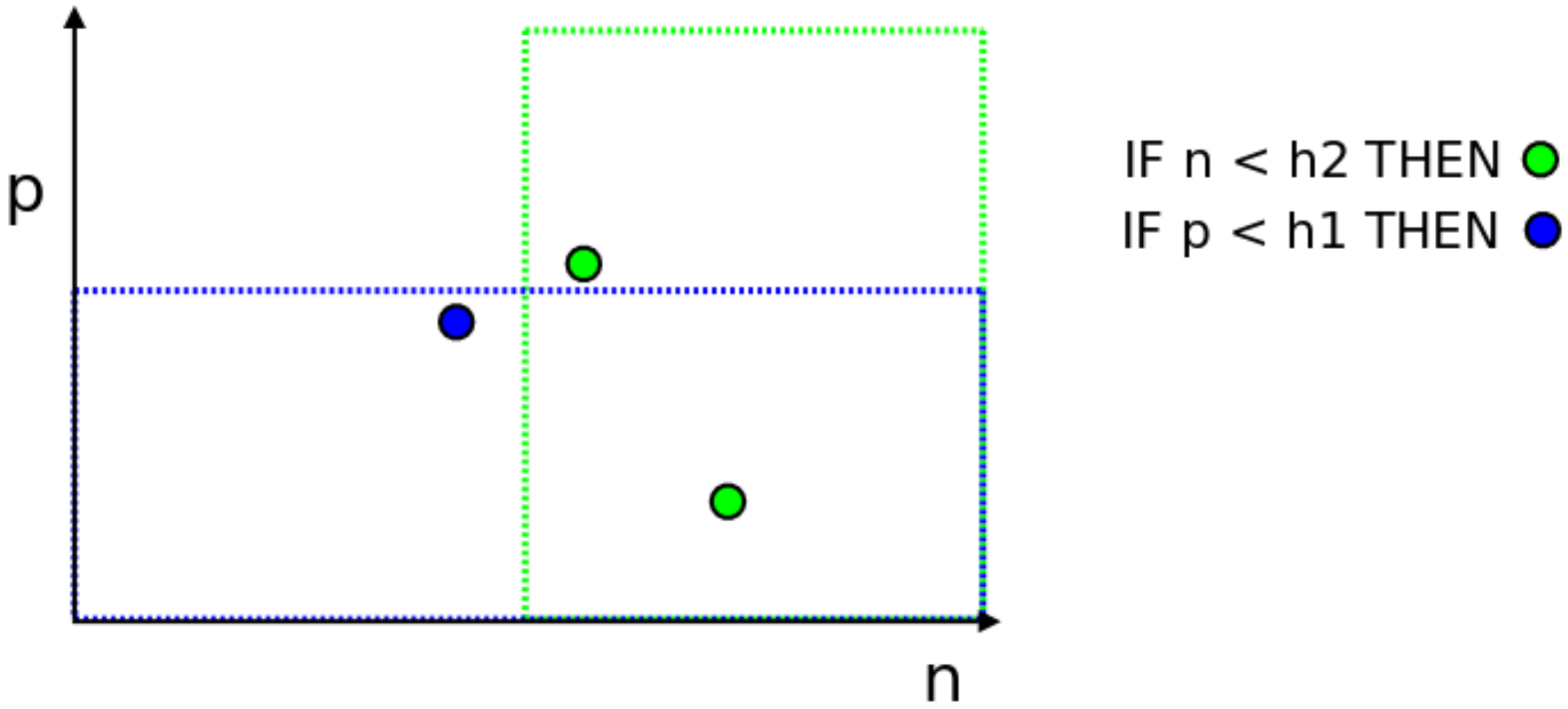

Modře, protože modré pravidlo je v seznamu dřív.

# Konflikt pravidel



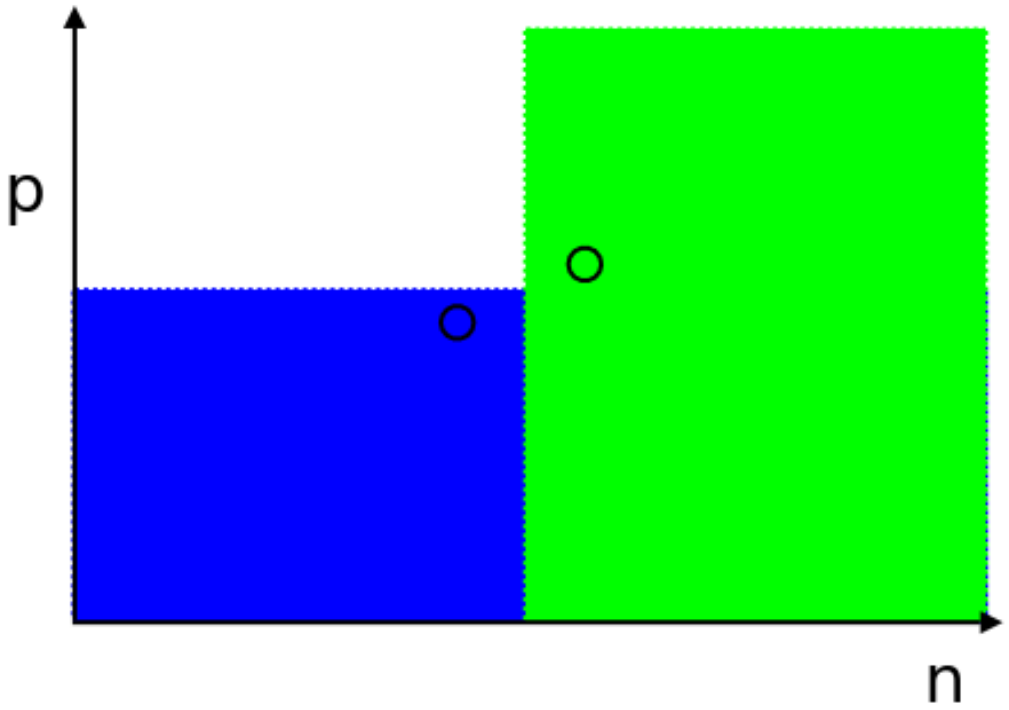

Modře, protože modré pravidlo je v seznamu dřív.

# Konflikt pravidel




Při obráceném pořadí pravidel klasifikujeme instanci zeleně.

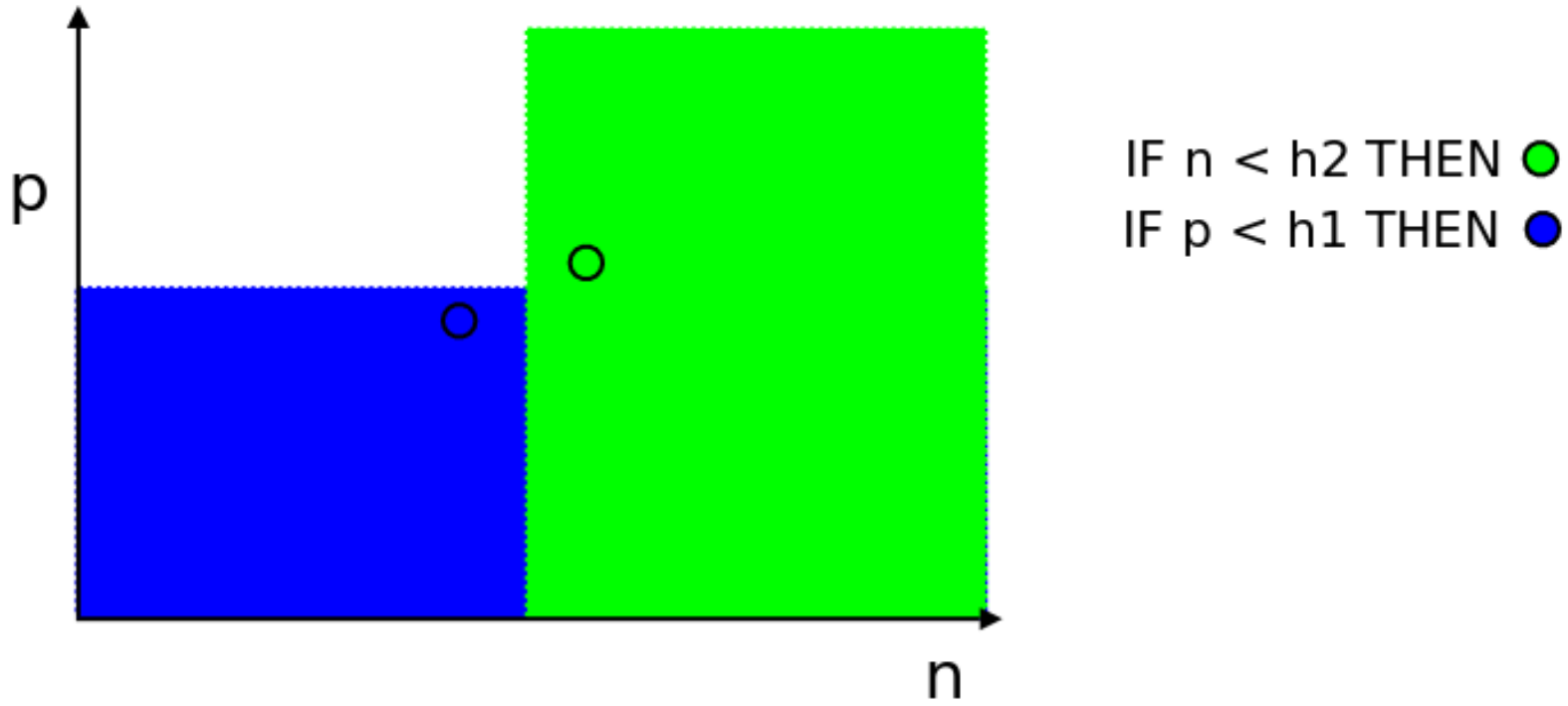
# Konflikt pravidel




IF n < h2 THEN ○
IF p < h1 THEN ●

Na pořadí pravidel záleží!

# Konflikt pravidel




Důsledek: implicitní pravidlo

**if** *true* **then** *většinová třída*

vždy na konec seznamu.

# Rozhodovací pravidla a asociace

- Hledání nejlepších podmínek rozhodovacího pravidla je vlastně hledání častých *asociací* v dané třídě instancí.

$$\textbf{if}\ \underbrace{\text{příjem=vysoký \& bydliště=praha}}_{\text{častá asociace}}\ \textbf{then}\ \underbrace{\textbf{splácí}}_{\text{v této třídě}}$$

- Asociace $\approx$ konjunkce současně platných podmínek
- Chceme, aby platila v co nejvíce instancích dané třídy a žádné instanci ostatních tříd.
- Co kdybychom jako třídu chápali *všechny* instance v datech?
- Cíl se zjednoduší: chceme asociace platné v co nejvíce instancích dat.

# Asociace

- Chceme najít asociace platné v co nejvíce instancích dat.
- Nejlepší řešení je triviální: prázdná konjuknce podmínek, tj. *true*.
- Platí ve všech instancích, ale není ničím zajímavá!
- Rozšiřme tedy zadání: chceme *všechny asociace* platné alespoň v $p\%$ instancí.
- Veličina $p$: tzv. **podpora** (support)

# Podpora asociace

Podpora: podíl instancí, v nichž asociace platí, mezi všemi instancemi

| **příjem** | **bydliště** | **pohlaví** | **úvěr** |
|---|---|---|---|
| vysoký | Praha | M | splácí |
| vysoký | Plzeň | M | splácí |
| nízký | Praha | M | nesplácí |
| vysoký | Praha | Ž | splácí |
| střední | Brno | M | splácí |

- Asociace $A =$

$$\text{příjem} = \text{vysoký} \ \& \ \text{bydliště} = \text{Praha}$$

  má podporu $2/5 = 0.4$.

- Zapisujeme

$$podp(A) = 0.4$$

# Podpora asociace

Všechny asociace s podporou alespoň 0.4

| **příjem** | **bydliště** |
|---|---|
| vysoký | Praha |
| vysoký | Plzeň |
| nízký | Praha |
| vysoký | Praha |
| střední | Brno |

# Podpora asociace

Všechny asociace s podporou alespoň 0.4

| příjem | bydliště |
|---|---|
| vysoký | Praha |
| vysoký | Plzeň |
| nízký | Praha |
| vysoký | Praha |
| střední | Brno |

1. *true*

# Podpora asociace

Všechny asociace s podporou alespoň 0.4

| příjem | bydliště |
|---|---|
| vysoký | Praha |
| vysoký | Plzeň |
| nízký | Praha |
| vysoký | Praha |
| střední | Brno |

1. *true*
2. příjem = vysoký

# Podpora asociace

Všechny asociace s podporou alespoň 0.4

| **příjem** | **bydliště** |
|---|---|
| vysoký | Praha |
| vysoký | Plzeň |
| nízký | Praha |
| vysoký | Praha |
| střední | Brno |

1. *true*
2. příjem = vysoký
3. bydliště = Praha

# Podpora asociace

Všechny asociace s podporou alespoň 0.4

| příjem | bydliště |
|---|---|
| vysoký | Praha |
| vysoký | Plzeň |
| nízký | Praha |
| vysoký | Praha |
| střední | Brno |

1. *true*
2. příjem = vysoký
3. bydliště = Praha
4. příjem = vysoký & bydliště = Praha

# Podpora asociace

Všechny asociace s podporou alespoň 0.4

| příjem | bydliště |
|---|---|
| vysoký | Praha |
| vysoký | Plzeň |
| nízký | Praha |
| vysoký | Praha |
| střední | Brno |

1. *true*
2. příjem = vysoký
3. bydliště = Praha
4. příjem = vysoký & bydliště = Praha

- Asociace, které mají požadovanou podporu, se nazývají **časté**.

# Analýza transakcí

- Hledání asociací se uplatňuje zejm. v *"analýze transakcí"* (*"analýze nákupních košíků"*)
- Příznaky: položky sortimentu. Instance: obsah nákupního košíku. Hodnoty příznaků $\{\text{ano}, \text{ne}\}$.

| **pivo** | **párky** | **horčice** | **pleny** |
|---|---|---|---|
| ano | ne | ne | ano |
| ne | ano | ano | ne |

- Zde místo např.

$$\text{pivo} = \text{ano} \;\&\; \text{pleny} = \text{ano}$$

- zapisujeme (a chápeme) asociaci jako *množinu položek*, např.

$$\{\text{pivo}, \text{pleny}\}$$

# Princip monotonicity

| pivo | párky | horčice | pleny |
|---|---|---|---|
| ano | ne | ne | ano |
| ano | ne | ano | ano |
| ne | ano | ano | ne |
| ano | ano | ano | ne |
| ano | ano | ano | ano |

# Princip monotonicity

| pivo | párky | horčice | pleny |
|---|---|---|---|
| ano | ne | ne | ano |
| ano | ne | ano | ano |
| ne | ano | ano | ne |
| ano | ano | ano | ne |
| ano | ano | ano | ano |

Pozorování:

- Není-li množina položek častá, pro $p = 0.5$ např.

$$\{\text{pivo}, \text{párky}, \text{hořčice}\}$$

# Princip monotonicity

| pivo | párky | horčice | pleny |
|---|---|---|---|
| ano | ne | ne | ano |
| ano | ne | ano | ano |
| ne | ano | ano | ne |
| ano | ano | ano | ne |
| ano | ano | ano | ano |

Pozorování:

- Není-li množina položek častá, pro $p = 0.5$ např.

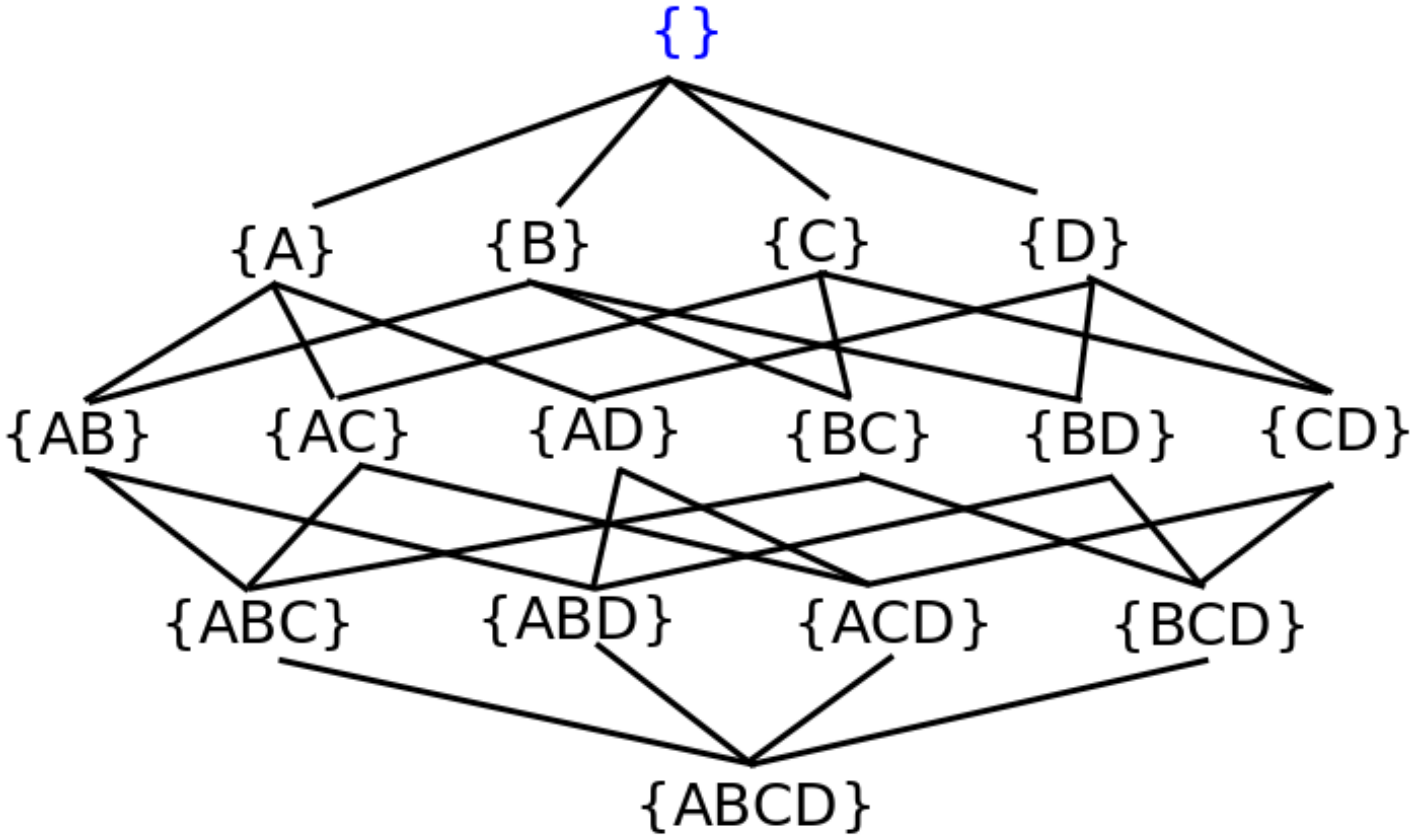
$$\{\text{pivo}, \text{párky}, \text{hořčice}\}$$

- není časté ani jakékoliv její rozšíření, např.

$$\{\text{pivo}, \text{párky}, \text{hořčice}, \text{pleny}\}$$

# Hledání častých množin položek algoritmem APRIORI

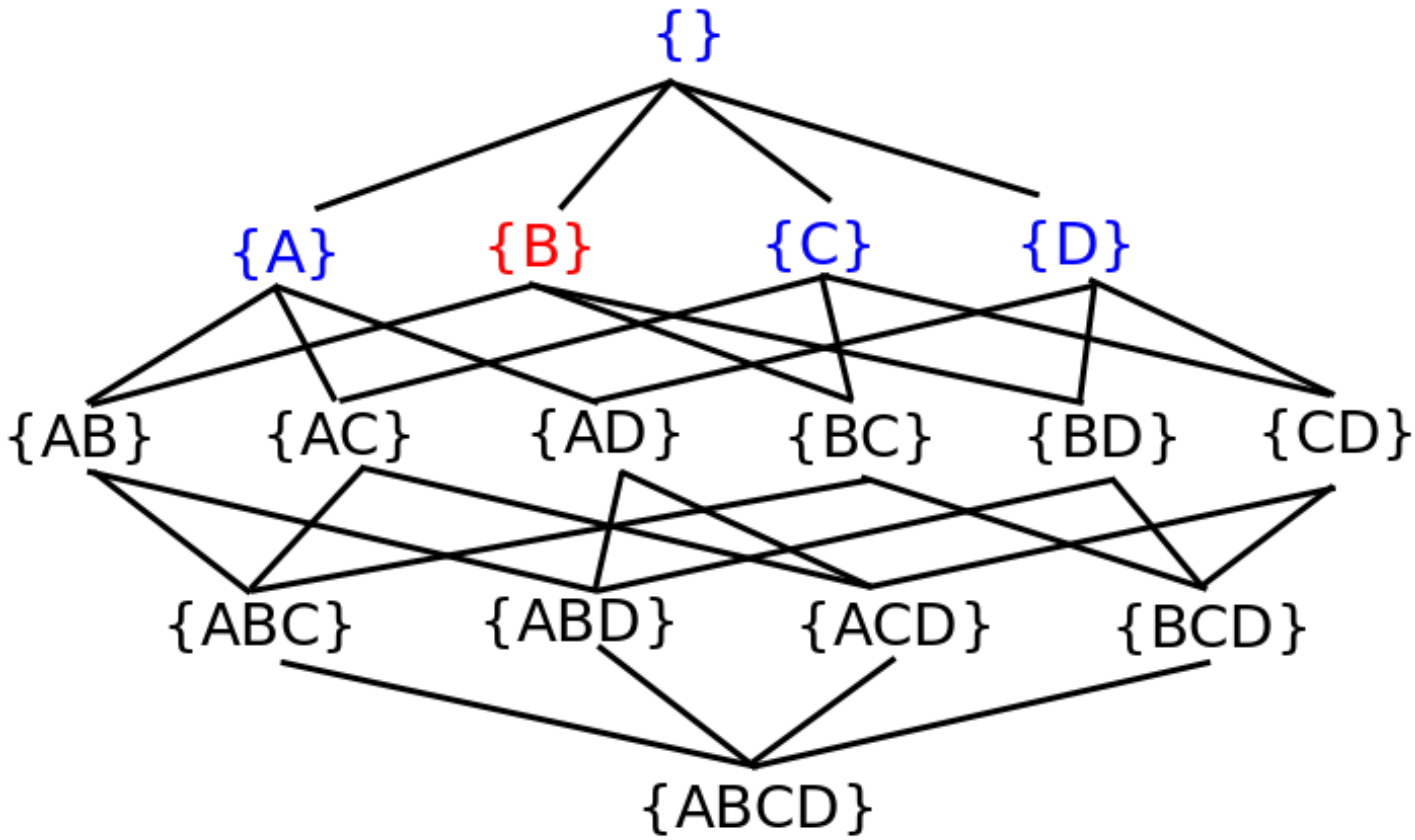
Postupujeme s vyšších pater do nižších (specializace).




Prázdná množina je vždy častá. Kandidáti o patro níž:
$\{A\}, \{B\}, \{C\}, \{D\}$.

# Hledání častých množin položek algoritmem APRIORI

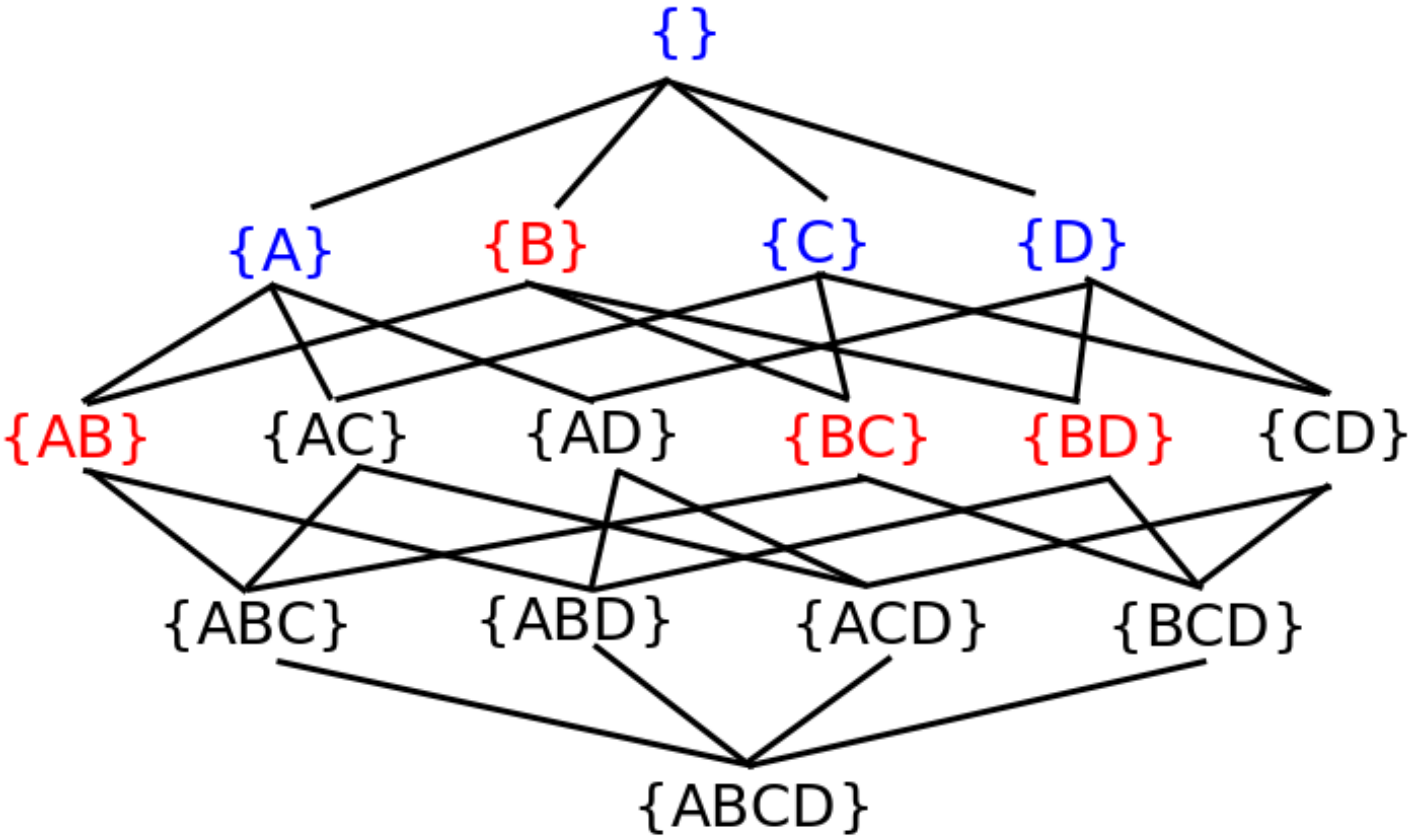
Postupujeme s vyšších pater do nižších (specializace).




V databázi zjistíme, že $\{B\}$ není časté, ostatní kandidáti ano.

# Hledání častých množin položek algoritmem APRIORI

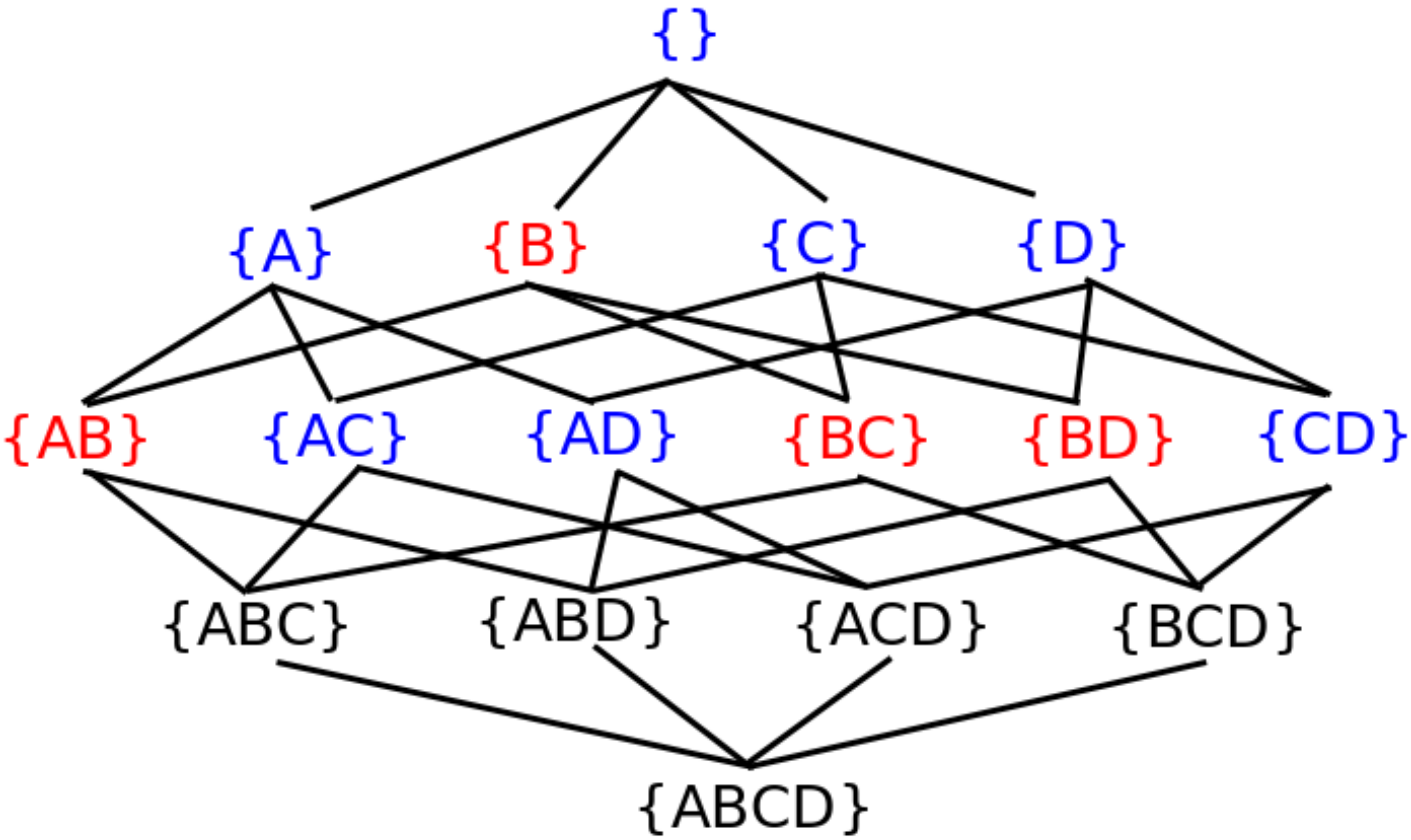
Postupujeme s vyšších pater do nižších (specializace).




$\{AB\}, \{BC\}, \{BC\}$ nemohou být časté (monotonicita).
Zbývají kandidáti $\{AC\}, \{AD\}, \{CD\}$.

# Hledání častých množin položek algoritmem APRIORI

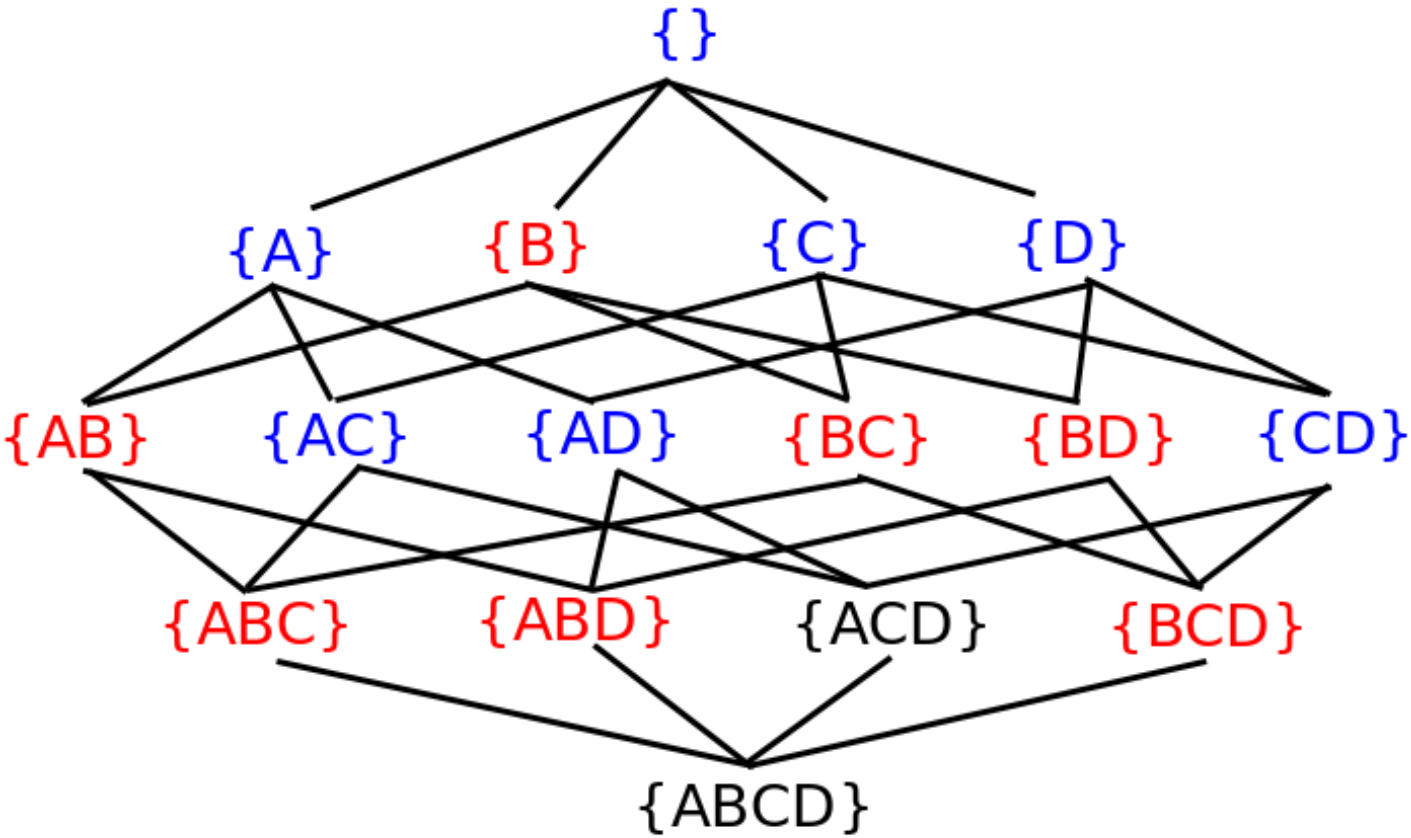
Postupujeme s vyšších pater do nižších (specializace).




V databázi zjistíme, že jsou všichni kandidáti častí.

# Hledání častých množin položek algoritmem APRIORI

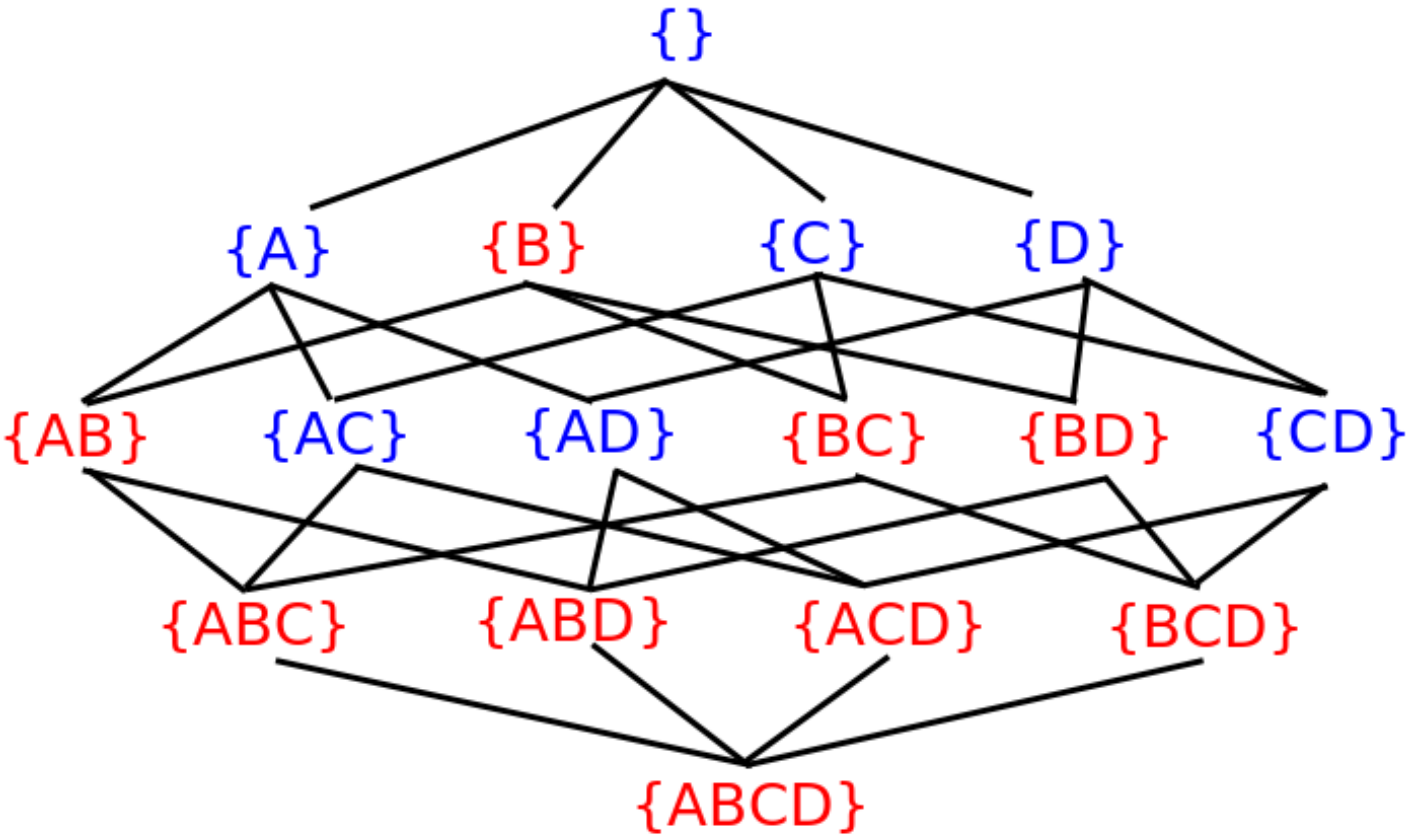
Postupujeme s vyšších pater do nižších (specializace).




V dalším patře už pouze kandidát $\{ACD\}$.

# Hledání častých množin položek algoritmem APRIORI

Postupujeme s vyšších pater do nižších (specializace).




V databázi zjistíme, že kandidát není častý. Žádné další časté množiny nejsou.

# Asociační pravidlo

- Pravidlo ve tvaru

$$\textbf{if } Ant \textbf{ then } Suc$$

kde $Ant$ a $Suc$ jsou množiny položek nazývané *antecedent* resp. *sukcedent*. Pravidlo též zapisujeme jako

$$Ant \rightarrow Suc$$

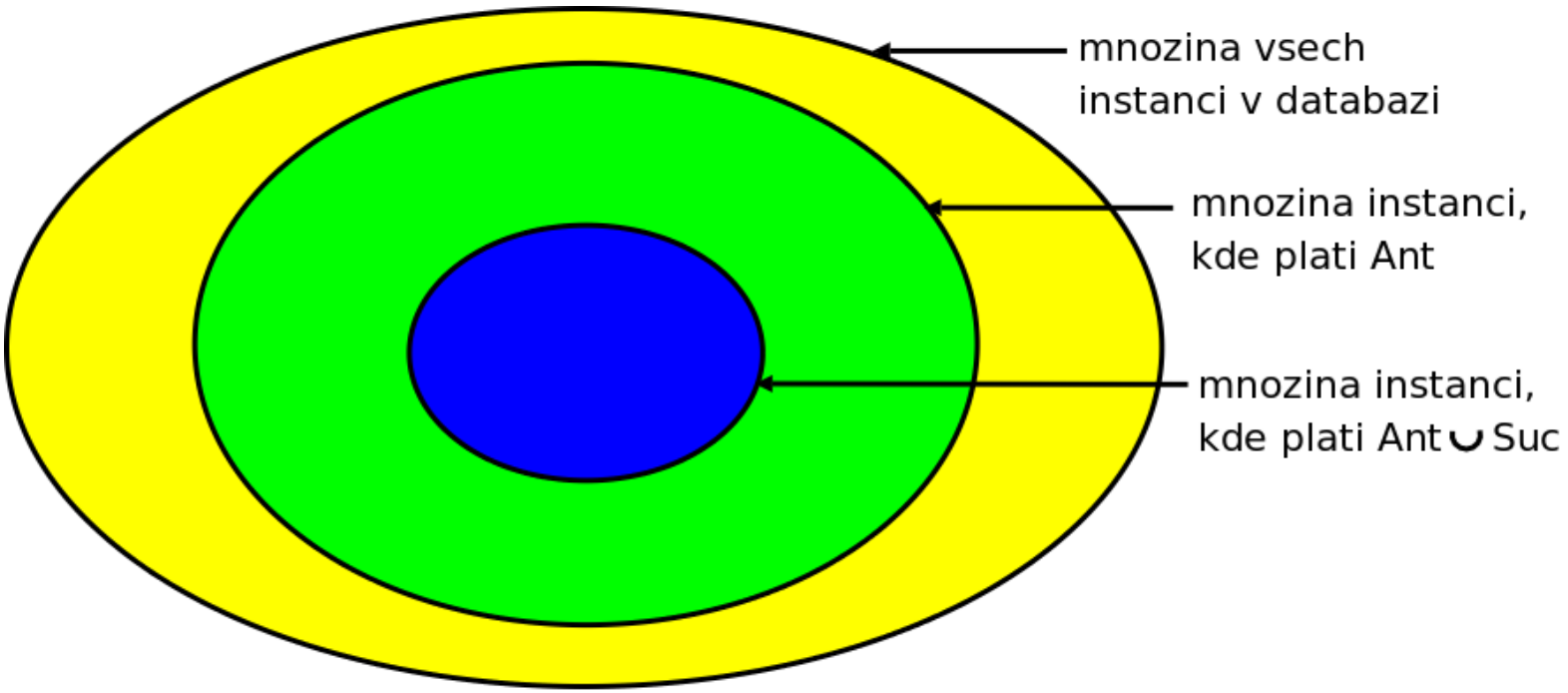
- **Podpora asociačního pravidla** $R$ definována jako

$$podp(Ant \rightarrow Suc) = podp(Ant \cup Suc)$$

- **Spolehlivost** (confidence) **asociačního pravidla** $R$ definována jako

$$spol(Ant \rightarrow Suc) = \frac{podp(Ant \cup Suc)}{podp(Ant)}$$

# Asociační pravidlo



# Hledání asociačních pravidel algoritmem APRIORI

- Hledáme pravidla $Ant \rightarrow Suc$, která jsou častá (tj. s podporou alespoň $p$) a spolehlivá (tj. se spolehlivostí alespoň $s$).
- Je-li pravidlo $Ant \rightarrow Suc$ časté, tak množina položek (asociace) $Ant \cup Suc$ je častá.
- Nejprve tedy najdeme všechny časté množiny položek.
- Pro každou častou množinu položek $X$ a každou její neprázdnou vlastní podmnožinu $M \subset X$ zkusíme, zda

$$\frac{podp(X)}{podp(M)} \geq p$$

- Pokud ano, tak pravidlo

$$M \rightarrow M \setminus X$$

je časté a spolehlivé.

# Hledání asociačních pravidel: příklad

- Hledáme asociační pravidla s podporou alespoň 0.1 a spolehlivostí alespoň 0.6. Právě 13% nákupních košíků obsahuje každou položku z množiny

$$\{\text{párky}, \text{hořčice}, \text{pivo}, \text{otvírák}\}$$

- Množina je tedy častá. Vyberme nějakou její vlastní neprázdnou podmnožinu, např.

$$\{\text{párky}, \text{hořčice}\}$$

- Pravidlo

$$\{\text{párky}, \text{hořčice}\} \rightarrow \{\text{pivo}, \text{otvírák}\}$$

má podporu 13%. Pokud navíc právě 19% nákupních košíků obsahuje párky i horčici, má pravidlo spolehlivost

$$\frac{13}{19} > 0.6$$

a je tedy časté i spolehlivé.