TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií

Časové řady v data miningových úlohách

Diplomová práce

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: 1802T007 – Informační technologie

Autor práce: **Bc. Jiří Kratochvíl**
Vedoucí práce: RNDr. Klára Císařová, Ph.D.

Liberec 2016

TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies

Time series in data mining tasks

Diploma thesis

Study programme: N2612 – Electrotechnology and informatics
Study branch: 1802T007 – Information technology

Author: **Bc. Jiří Kratochvíl**
Supervisor: RNDr. Klára Císařová, Ph.D.

Liberec 2016

Technická univerzita v Liberci
Fakulta mechatroniky, informatiky a mezioborových studií
Akademický rok: **2016/2017**

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Jiří Kratochvíl**
Osobní číslo:	**M14000167**
Studijní program:	**N2612 Elektrotechnika a informatika**
Studijní obor:	**Informační technologie**
Název tématu:	**Časové řady v data miningových úlohách**
Zadávající katedra:	**Ústav mechatroniky a technické informatiky**

Zásady pro vypracování:

1. Seznamte se s problémy spojenými s analýzou časových řad.
2. Prostudujte použití časových řad v data miningových úlohách.
3. Zpracujte případovou studii s využitím dat představujících časové řady (časové sekvence), a to v IBM SPSS Modeleru a KNIME.
4. Srovnejte oba nástroje a vytvořte doporučení jejich nasazení pro tento typ úloh.

Rozsah grafických prací: **dle potřeby dokumentace**

Rozsah pracovní zprávy: **40–50 stran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

[1] **Arlt J., Arltová M.: Ekonomické časové řady, Professional Publishing, 2009**

[2] **Litschmannová M.: Úvod do analýzy časových řad, VŠB Ostrava, 2010**

[3] **Yong Yin, Ikou Kaku, Jiafu Tang: Data Mining, Springer London Ltd, 2011**

[4] **Olivia Parr Rud: Data mining, Computer Press, a.s., 2006**

Vedoucí diplomové práce: **RNDr. Klára Císařová, Ph.D.**
Ústav mechatroniky a technické informatiky

Datum zadání diplomové práce: **10. října 2016**

Termín odevzdání diplomové práce: **15. května 2017**

Technická univerzita v Liberci

L.S.

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2016

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 3.1.2017

Podpis:

Abstrakt

Data mining je v současné době mocným nástrojem pro analýzu dat. Pomocí této techniky lze objevovat skryté informace, předpovídat vývoj nemocí, nebo monitorovat průběh výroby, který bývá reprezentován formou časových řad. Diplomová práce je zaměřena na popis a členění časových řad spolu se zpracováním případových studií založených na časových řadách. V práci byly zkoumané a popsané případové studie v data miningových nástrojích používaných na fakultě mechatroniky pro obor informační technologie v kurzu data miningu a jsou zachycovány rozdíly při jejich používání. V závěru jsou nástroje vyhodnoceny spolu s doporučením z hlediska použití v dalších úlohách.

Abstract

Data mining is currently a powerful tool for analyzing data. This technique can discover a hidden information, predict a development of dieseases or monitor a process of production which is represented in many cases by form of time series. In this thesis were examined and reported case studies using data mining tools used by the faculty of mechatronics in the field of information technology in the data mining course and described differences in their use. Data mining tools are evaluated with the recommendations in terms of use in other tasks.

Poděkování

Mnohokrát děkuji RNDr. Kláře Císařové, Ph.D. za trpělivost, ochotu a poskytnuté rady, bez kterých by tato diplomová práce nemohla vzniknout. Práci bych chtěl věnovat Ing. Miloši Sedláčkovi za všechno, co pro mne v životě udělal.

Obsah

1 Úvod

V současné době se v řadě odvětví elektronicky zaznamenává obrovské množství dat na rozdíl od dřívějších dob, kdy záznam dat nebyl v takovém měřítku. Nastává tedy otázka, jak nasbíraná data smysluplným způsobem využít. Odpovědí na tuto otázku je ***data mining***, do češtiny často překládáno jako *dolování dat*. *Data mining* slouží k nalezení a zkoumání skrytých a užitečných informací ze zkoumaných dat. Díky této technice je možné z dat předvídat nadcházející trendy, vývoj nemocí, předpovídat budoucí prodeje nebo plánovat výrobu. Data z oblasti, jakou je například výroba, jsou často reprezentovány ve formě časových řad.

Diplomová práce se zabývá vytvořením případových studií obsahující časové řady spolu se sestavením predikčních modelů modelovaných v data miningových nástrojích. Použité data miningové nástroje byly vybrány na základě studijního využití ve výuce a to z důvodu, že diplomová práce má za cíl vytvořit případové studie pro výuku praktického zpracování časových řad na fakultě mechatroniky v předmětu zabývajícím se data miningem.

Před samotnou tvorbou případových studií je zapotřebí uvést čtenáře do problematiky časových řad a jejich členění. V kapitole 2 se zabývám představením základních pojmů časových řad, jejich charakteristikami a v poslední řadě manipulací s časovými řadami. Kapitola 3 je věnována úvodu čtenáře do problematiky data mimingu. V textu je popsána příprava dat před nasazením data miningových nástrojů spolu s představením vybraných modelovacích nástrojů, které mají v dnešní době velké zastoupení.

Stěžejní část diplomové práce je zaměřena na vytvoření případových studií založených na časových řadách. První případová studie, kterou jsem označil jako *monitorování provozu strojů*, je založena na predikci poruchy strojů za pomocí vytvořeného modelu zpracovávajícího poskytnuté data ve formě časové řady zaznamenávající vlastnosti stroje jako je tlak, teplota, výkon aj. před jeho selháním. Druhá případová studie je označena jako *předpověď spotřeby elektrické energie*. Studie je zaměřena na analýzu a predikci spotřeby elektrické energie na následující den pomocí naměřených časových řad spotřeby elektrické energie za dané období spolu s teplotou na několika stanovištích. Studie dále obsahuje analýzu spotřeby elektrické energie v závislosti na dnu v týdnu, svátku a venkovní teploty. V obou případech jsem vytvořil detailní příručky zachytávající nutné operace pro tvorbu případových studií ve zvolených nástrojích. Příručky také slouží jako podklad pro výuku.

Zpracování případových studií probíhá v data miningových nástrojích ***IBM SPSS Modeler*** a nástroji ***Knime***. Výběr nástrojů byl založen za cílem porovnání komerčního nástroje (*IBM SPSS Modeler*) spolu s volně dostupnou platformou *Knime*. Cílem je porovnat oba použité nástroje při modelování případových studií z hlediska nabízených funkcí a vyvodit doporučení pro nasazení na konkrétní typy úloh.

2 Analýza časových řad

Časové řady se vyskytují v každém odvětví, kde jsou produkována a zaznamenávána data v závislosti na čase, a jsou mocným nástrojem pro nalezení určitých, opakujících se událostí. Takové jevy se dají objevit pomocí analytických metod. Z důvodu velkého rozsahu analytických metod bude v této diplomové práci popsána pouze vybraná část. Cílem analyzačních metod je podle [1] následující: Metody pro analýzu časových řad slouží k nalezení pravidla pro tvorbu časové řady. Nalezením pravidla se dostáváme do situace, kdy máme šanci předvídat budoucí vývoj zkoumané časové řady a přizpůsobit podle ní další činnost, která je na časové řadě závislá. Po nalezení mechanismu vytváření časové řady se vytváří model. Model se následně testuje a ověřuje se jeho korektnost vůči zkoumané časové řadě.

V této kapitole budou nejprve vymezeny některé základní termíny, které čtenáře uvedou do zkoumané problematiky. Dále budou popsány vlastnosti časových řad spolu s jejich možnostmi analýzy.

2.1 Časové řady

Časová řada je posloupnost časově uspořádaných hodnot. Hodnoty jsou nejčastěji výsledkem měření, nebo sledováním veličin jakou jsou ekonomické ukazatele apod. Časové řady mohou obsahovat například data o stavu oleje výrobního stroje sledované v čase, nebo stavy zásob skladů za dané období. Časové řady bývají nejčastěji získávány ekvidistantně, tzv. jsou získány ve stejných časových intervalech. Příkladem ekvidistantních intervalů je periodické zaznamenávání teploty vzduchu každou hodinu.

Zápis časové řady je v následujícím tvaru:

$$y_1, y_2, \ldots, y_n \tag{2.1}$$

nebo -li

$$y_t, t = 1, \ldots, n \tag{2.2}$$

Časové řady slouží k vytvoření modelu, podle kterého můžeme např. predikovat některé situace. [1]

Situace, které lze z časových řad předpovídat jsou například predikce nezaměstnanosti, havárie strojů, nebo intervence na devizových trzích.

2.2 Dělení časových řad

Časové řady popisují vývoj statistického znaku a rozdělují se do mnoha skupin [4]. Mezi hlavní skupiny dělení patří řady:

Intervalová časová řada

Hodnota sledovaného ukazatele časové řady závisí na celé délce sledovaného intervalu. Příkladem může být růst mezd v České republice od roku 2000 do roku 2015, tzv. o kolik Kč se změnila mzda od roku 2000 do roku 2015. Dále pak například měsíční provozní náklady pro chod restaurace nebo čtvrtletní spotřeba vody pro sledované sídliště.

Okamžiková časová řada

Hodnota ukazatele okamžikové časové řady nezávisí na délce sledovaného intervalu, jako v případě intervalové časové řady, nýbrž na jistém okamžiku. Tímto okamžikem může být určitý čas, den v měsíci apod. Příkladem je počet dopravních nehod k datu 31.12.2014, nebo počet evidovaných uchazečů v určitém okamžiku apod.

Odvozená časová řada

Poslední rozdělení časových řad je odvozená časová řada, která je výsledkem kombinace řad intervalových nebo řad okamžikových. Příkladem může být efektivita výroby. Výsledná časová řada bude vytvořena podílem řady, která reprezentuje počet kusů výrobku a řady reprezentující nezávadné výrobky.

2.3 Další dělení časových řad

Časové řady lze dále dělit do několika skupin [1, 3, 4] a to podle délky časových řad, ekvidistance, zda jsou deterministické či nikoliv apod. Dělení řad podle zmíněných kritérií jsou popsány v následujících odstavcích.

Řady s absolutními a relativními ukazateli

Hodnoty časové řady s absolutními ukazateli nejsou žádným způsobem upraveny, ale jsou ve „*stavu*“, ve kterém byly získány (naměřeny). Naproti tomu časové řady s relativními ukazateli jsou již nějakým způsobem upraveny. Ukazatelé mohou být zprůměrovány, mohou být nad těmito ukazateli prováděny různé výpočty apod.

Řady deterministické a stochastické

Deterministické časové řady neobsahují žádný náhodný prvek a jsou tvořeny podle určitého modelu. Naopak u stochastických řad se náhodný prvek vyskytuje a nelze je tak predikovat, nýbrž pouze odhadovat. Příkladem stochastické časové řady je vývoj akcií na akciovém trhu, který se odvíjí podle situací ve světě.

Řady krátkodobé a dlouhodobé

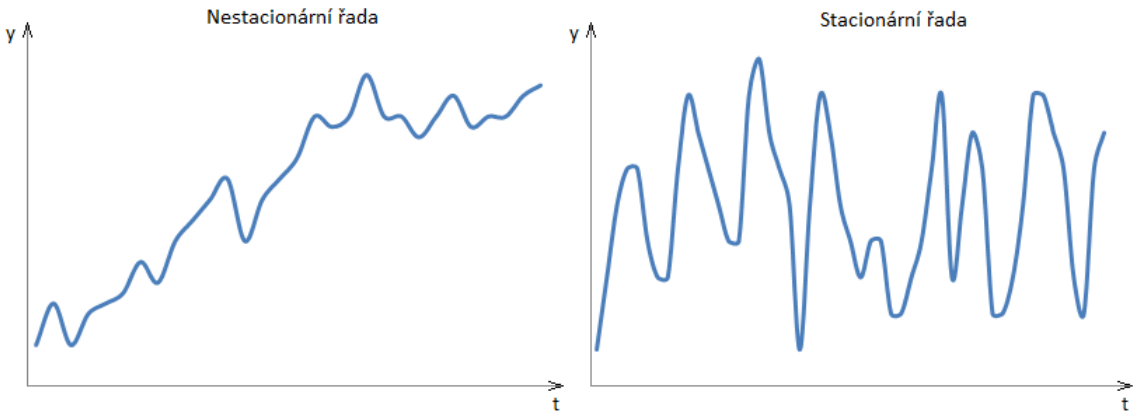
Hodnoty krátkodobých časových řad jsou zaznamenávány v časových úsecích kratších než jeden rok. Jedná se například o měsíční uzávěrky, čtvrtletní hodnocení apod. U dlouhodobých časových řad se hodnoty sledují v horizontu větším než jeden rok. Takové řady například obsahují hodnoty o subjektu za posledních x let a obsahují tak historii.

Řady ekvidistantní a neekvidistantní

Ekvidistantní řady obsahují hodnoty se stejným časovým krokem, tzv. interval mezi hodnotami řady je stejný a hodnoty jsou zaznamenávány ve stanovém čase. Neekvidistantní časové řady žádným intervalem nedisponují a hodnoty řad jsou získávány náhodně, nebo v závislosti na okolních procesech. Při práci s neekvidistantními řadami je nutné nejprve tyto řady upravit. Často se uvádí převod na jednotkový interval. Ekvidistantní časovou řadu lze získat například periodickým snímáním teploty ovzduší každou hodinu. Neekvidistantní časovou řadu lze pořídit zaznamenáním hodnot při selhávání elektroniky a to měřením např. napětí a proudu na klíčových elektronických součástkách.

Řady stacionární a nestacionární

Označení řady pojmem stacionární a nestacionární časové řády závisí na tom, zda obsahují změny v průměru, nebo rozptylu hodnot řady. Stacionární řady obsahují cyklické změny průměru nebo rozptylu a tudíž nejsme schopni rozeznat jednotlivé hodnoty v různých časových okamžicích, tzv. dochází k oscilování hodnot. Naopak u nestacionárních časových řad dochází k vývoji hodnot na celém časovém měřítku řady. Nestacionární řady obsahují trend. Stacionární řadou může být například diskrétní sinusový signál. Příkladem nestacionární časové řady je vývoj světové populace, která se neustále zvyšuje. Rozdíl mezi těmito řadami je zobrazen na následujícím obrázku.



Obrázek 2.1: Rozdíl mezi nestacionární a stacionární časovou řadou

2.4 Charakteristiky časových řad

S časovými řadami je možné provádět mnoho matematických operací, které se ve statistice používají. Operace nad časovými řadami se dělí do dvou skupin. První skupinu tvoří charakteristiky polohy, kde hlavní skupinu tvoří průměry. Druhou skupinu zastupuje rozptyl, který je jedním ze zástupců charakteristiky variability. Mezi základní operace s časovými řadami jsou zmíněný aritmetický průměr a rozptyl. [3, 4] Tyto a další nejvíce používané operace nad časovými řadami jsou popsány v následujícím textu.

2.4.1 Charakteristiky polohy

Jedná se o hodnoty, které charakterizují střed zkoumaného souboru a kolem kterých kolísají všechny hodnoty souboru.

Aritmetický průměr

Aritmetický průměr je podíl součtu všech hodnot y_i časové řadu celkovým počtem prvků řady n.

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n} \tag{2.3}$$

Vážený průměr

Výsledkem váženého průměru je podíl součtu hodnot y_i časové řady, které jsou vynásobeny svoji váhou w_i, a celkovým součtem všech vah w_i

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i w_i}{\sum_{i=1}^{n} w_i} \tag{2.4}$$

Chronologický průměr

Chronologický průměr se používá pro zjištění průměrné hodnoty okamžikové časové řady.

Pro ekvidistantní časové řady se používá prostý chronologický průměr.

$$\bar{y} = \frac{\frac{y_1+y_2}{2} + \frac{y_2+y_3}{2} + \ldots + \frac{y_{n-1}+y_n}{2}}{n-1} = \frac{\frac{1}{2}y_1 + \sum_{i=2}^{n} y_i + \frac{1}{2}y_n}{n-1} \tag{2.5}$$

,kde y_i jsou hodnoty časové řady.

Pro neekvidistantní řady se používá následující vztah, tzv. vážený chronologický průměr.

$$\bar{y} = \frac{\frac{y_1+y_2}{2}d_2 + \frac{y_2+y_3}{2}d_3 + \ldots + \frac{y_{n-1}+y_n}{2}d_n}{d_2 + d_3 + \ldots + d_n} \tag{2.6}$$

nebo-li

$$\bar{y} = \frac{\sum_{i=2}^{n} \frac{y_{i-1}+y_i}{2} d_i}{\sum_{i=2}^{n} d_i} \tag{2.7}$$

,kde d_i; $i = 2, \ldots, n$ je délka intervalů mezi jednotlivými hodnotami časové řady.

2.4.2 Charakteristiky variability

Charakteristiky variability slouží ke zjištění variability zkoumaného souboru, tj. o kolik se liší zkoumané hodnoty v celé řadě.

Rozptyl

Udává, jak jsou hodnoty rozptýleny od střední hodnoty.

$$Var\left(X\right) = \frac{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2}{n-1} \tag{2.8}$$

kde $n-1$ značí počet prvků řady výběrového souboru, y_i je prvek řady a $\bar{y}$ je průměr hodnot řady.

Směrodatná odchylka

Jedná se o druhou odmocninu rozptylu a značí, jak jsou hodnoty odchýleny od průměru.

$$\sigma = \sqrt{Var\left(X\right)} \tag{2.9}$$

2.5 Korelace

Korelace vyjadřuje míru závislosti dvou časových řad. Korelace je dána vztahem

$$r_{xy} = \frac{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sigma_x \sigma_y} \tag{2.10}$$

, kde hodnoty r_{xy} mohou nabývat hodnot z intervalu $\langle -1; 1\rangle$. Veličiny x_i a y_i jsou hodnotami zkoumaných řad a σ_x, σ_y směrodatné odchylky časových řad x a y. Pokud se hodnoty r_{xy} výsledku korelace blíží k hodnotě -1, dochází k záporné korelaci. Hodnoty jedné časové řady v záporné korelaci stoupají, kdežto hodnoty v druhé časové řadě klesají. Pokud korelace časových řad nabývá hodnoty 1, tak se hodnoty korelujících řad vyvíjejí stejně (hodnoty rostou). Korelace nabývají hodnoty 0 udává nezávislost zkoumaných řad.

Pokud se však u časových řad vyskytuje časové zpoždění, tj. hodnoty časových řad jsou získávány v jiných časových intervalech, pak závislost těchto řad nemusí být odhalena. Časové řady obsahující časové zpoždění by měli být nejdříve upraveny do stejných časových intervalů. [4]

2.6 Míry dynamiky

Mezi další transformace časových řad patří míry dynamiky. Míry dynamiky zahrnují tempa a přírůstky. Tempem se rozumí procentuální růst sledované veličiny v daném období oproti období minulému. Přírůstek pak označuje přírůstek sledovaného ukazatele mezi dvěma následujícími obdobími. Popis jednotlivých mír dynamiky vychází z [1, 3].

Absolutní přírůstek

Absolutní přírůstek označuje rozdíl současného a minulého období , tj. t, $t-1$ sledované veličiny.

$$\Delta y_t = y_t - y_{t-1} \tag{2.11}$$

kde $t = 2, 3, \ldots$

Průměrný absolutní přírůstek

Jedná se průměrnou hodnotu absolutních přírůstků za stanovené období n
kde $n = 2, 3, \ldots$

$$\bar{\Delta} = \frac{\sum_{t=2}^{n} \Delta y_t}{n-1} \tag{2.12}$$

Koeficient růstu

Koeficient růstu označuje podíl dvou sousedních období, tj. o kolik procent vzrostla sledovaná veličina o proti období minulému. Koeficient růstu se také označuje jako tempo růstu. Pro získání procent se výsledek násobí hodnotou 100.

$$k_t = \frac{y_t}{y_{t-1}} \tag{2.13}$$

kde $t = 2, ..., n$

Průměrný koeficient růstu

Označuje průměrný koeficient růstu za stanovené období n.

$$\bar{k} = \sqrt[n-1]{k_2 \cdot k_3 \cdot ... \cdot k_n} = \sqrt[n-1]{\frac{y_2}{y_1} \cdot \frac{y_3}{y_2} ... \cdot \frac{y_n}{y_{n-1}}} \tag{2.14}$$

nebo-li

$$\bar{k} = \sqrt[n-1]{\frac{y_n}{y_1}} \tag{2.15}$$

Relativní přírůstek

Udává změnu sledovaného ukazatele časové řady v čase t oproti hodnotě ukazatele v čase $t-1$.

$$\sigma_t = \frac{\Delta y_t}{y_{t-1}} = \frac{y_t - y_{t-1}}{y_{t-1}} \tag{2.16}$$

2.7 Úpravy časových řad

V následujících odstavcích budou popsány základní úpravy časových řad.

Doplnění chybějících hodnot

V některých případech je nutné časovou řadu doplnit o určitý počet hodnot z důvodu, že zkoumaná řada je pro další práci příliš *„krátká“*. V takovémto případě je několik způsobů, jak tyto chybějící hodnoty doplnit [4]:

- První z možností je doplnit řadu nulami. Doplněním lze zvětšit velikost časové řady na požadovanou velikost. Jedná se o nejjednoduší metodu, ale nemá žádnou vypovídající hodnotu z důvodu nepřidání žádné nové informace.
- Další možnost doplnění chybějících hodnot je doplnit řadu aritmetickým průměrem nebo mediánem. Tato metoda je často používána a využívá se především pro doplnění hodnot mezi hodnoty již naměřené.
- Mezi další metody doplnění hodnot patří metoda lineární interpolace. Lineární interpolace vkládá mezi jednotlivé sousední hodnoty, hodnoty nové získáním lineární interpolací.
- Pro doplnění hodnot lze také použít metodu extrapolace. Extrapolace slouží k odhadu hodnot řady mimo tuto řadu. Extrapolované hodnoty mají v praxi smysl pouze pro malé intervaly. Pro větší intervaly je extrapolace považována za velmi nepřesnou metodu.

Doplňováním časových řad chybějícími hodnotami je v praxi nutné brát zřetel nad tím, že dochází k dosazováním *„uměle“* vytvořeným hodnotám, které by v praxi nemuseli vůbec nastat.

Transformace měřítka

Transformace měřítka se používá pro zmenšení variability hodnot časové řady. Transformace může být uskutečněna za pomocí matematických operací, a to například vynásobením řady vhodnou konstantou, logaritmováním, kombinací dvou řad apod. Pro získání původní řady se provede zpětná transformace. [4]

Sezónní diference

Sezónní diference slouží ke zjištění, zda došlo k diferenci v opakujících se (sezónních) okamžicích. Diference může být kladná nebo záporná. Příkladem je diference čtvrtletních uzávěrek za předešlé období několika let, nebo změna vyrobených produktů v určitém měsíci za několik let. [4]

Vyhlazování časových řad

Časové řady mohou někdy obsahovat několik náhodných hodnot časové řady, nebo hodnoty, které se do časové řady mohou dostat z důvodu chybného měření. Vyhlazení časové řady je schopné tyto jevy částečně potlačit a zobrazit tak řadu bez dalších vlivů. Vyhlazení časové řady lze docílit pomocí klouzavých průměrů. [4]

2.8 Problémy časových řad

Při tvorbě časové řady se sledovaná veličina transformuje na matematický model. Transformací může dojít k řadě problémům, kterým je vhodné se věnovat před vlastní analýzou časové řady [1, 4].

Problémy s délkou časových řad

U časových řad může docházet k problémům z důvodu délky řady. Například některé algoritmy pro analýzu časové řady požadují minimální velikost časové řady pro správné fungování algoritmu. Další problém *„krátkých“* řad je, že řady nemusí dostatečně zachytit sledovaný vývoj zkoumaného problému. V dlouhých řadách se naopak může vyskytovat prvek, který ovlivní řadu a znehodnotí očekávaný výsledek pozorování.

Problémy s kalendářními dny

Dalším problémem časových řad je problém kalendářních dnů. Problém spočívá v délce jednotlivých měsících, ale také v pracovních dnech. Pracovní dny jednoho měsíce často nekorespondují s pracovními dny v jiném měsíci. Nelze zapomenout na svátky nebo přestupný rok, které mohou ovlivňovat časové řady a komplikovat související algoritmy. Problémy kalendářních dní lze řešit stanovením pevného počtu dní měsíce, roku apod.

Problémy se vzorkem měření

Výběr vzorku naměřených dat časové řady nemusí být vždy prováděn stejným způsobem nebo vybraný vzorek nemusí korektně reprezentovat zkoumanou časovou řadu. Například pouhým zkoumáním několika počátečních záznamů časové řady nezjistíme chování časové řady a získaný vzorek nemusí korespondovat s dalším vývojem. Výsledná analýza vzorkované časové řady by byla velmi zkreslená.

Problémy s volbou časových bodů pozorování

Při tvorbě časové řady se sledovaná veličina spojitého charakteru přeměňuje na řadu diskrétních hodnot, které jsou časově uspořádány. Při transformaci je nutné zvolit způsob diskretizace. Základní technikou diskretizace je zaznamenávání hodnoty v konkrétním čase, například počet bitů přenesených v konkrétním okamžiku. Další

metodou diskretizace je akumulace hodnot. Akumulace hodnot sčítá jednotlivé hodnoty časové řady za určité období. Poslední část diskretizace je metoda průměrování. Metoda při diskretizaci spojité veličiny průměruje diskrétní hodnoty za zvolené období.

2.9 Metody analýzy časových řad

Pro analýzu časových řad existuje řada metod, které se liší způsobem aplikace na daný problém. Metody jsou dále závislé na aplikaci na konkrétních časových řadách. Mezi nepoužívanější metody patří [4, 1]:

- Grafická analýza
- Dekompozice časových řad
- Spektrální analýza
- Box-Jenkinsovská metodologie

Grafická analýza

Jedná se o základní metodu analýzy časové řady. Jak již název napovídá, při této analýze se zkoumá časová řada reprezentována grafem. Proložením grafu lze například zjistit přibližný trend časové řady nebo vyhledáváním tzv. ***patterns***[1]lze odhadovat budoucí vývoj. Grafická analýza se například využívá při analýze akcií. Použitím grafické analýzy jsme schopni porovnávat více časových řad mezi sebou a hledat určité závislosti.

Dekompozice časových řad

Dekompozice časových řad je metoda umožňující časovou řadu rozdělit do několika složek. Důvodem rozdělení časové řady na jednotlivé složky je její snadnější následná analýza, než analyzovat časovou řadu jako celek. Princip metody bude popsán v následující kapitole 2.10.

Spektrální analýza

Spektrální analýzu lze přirovnat k fourierově analýze. Časová řada je považována za kombinaci sinusových a kosinusových funkcí, které mají různé amplitudy a frekvence. Princip metody spočívá v analýze spektra, tedy nalezení frekvencí, které řada zastupuje a následná analýza jednotlivých složek spektra.

[1] Vzory, které se mohou vyskytovat v časových řadách a je možné z nich odvodit budoucí vývoj řady

Box-Jenkinsovská metologie

Box-Jenkinsovské metologie staví na analýze reziduální (náhodné) složky, která je popsána v kapitole 2.10. Analýza zjišťuje možnou korelaci veličin reziduální složky s jinými prvky řady. Box-Jenkinsovská metologie zkoumá možnou závislost pomocí změny vstupních dat a pozorováním změny prvků zkoumané řady.

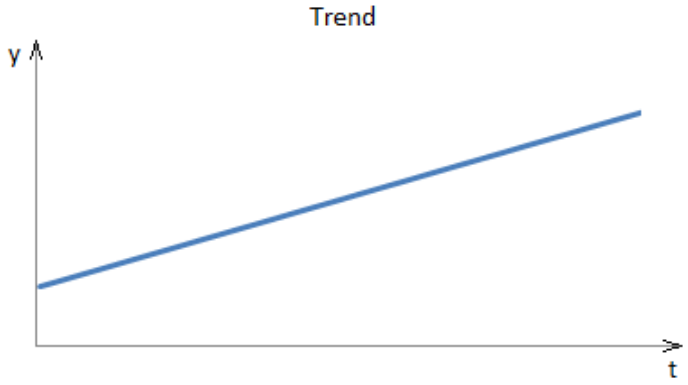
2.10 Dekompozice časových řad

Rozkladem časové řady na složky lze snadněji řadu zkoumat a odhalit různé vlastnosti vývoje zkoumané řady jako celku. Tyto složky vychází z potřeb klasické analýzy ekonomických časových řad a je možné hledat stejné analogie i v jiných časových řadách. Rozklad však klade za předpoklad, že model časové řady je závislý pouze na čase a není ovlivňován žádnými dalšími vlivy. Popis složek časové řady vychází z [1, 3, 4]

Předpokladem dekompozice časových řad je dekompozice časové řady do několika jednotlivých složek:

- Trend - T_t
- Sezonní složka - S_t
- Cyklická složka - C_t
- Náhodná složka - E_t

Trend T_t
Určuje směr vývoje zkoumaného jevu z dlouhodobého hlediska, tedy je výsledkem dlouhodobých a stálých procesů. Trend může být rostoucí, klesající, nebo konstantní (bez trendu). Trend se modeluje obvykle pomocí matematických křivek. Příkladem trendu mohou být životní podmínky v dané zemi, pokroky ve vědě nebo porodnost v ČR. Obrázek 2.2.



Obrázek 2.2: Trend

Sezónní složka S_t
Jedná o jevy, které se každoročně opakují a to vždy ve stejných obdobích. Příkladem sezónní složky je změna související s ročním obdobím, národní svátky, tradice apod. S těmito změnami jsou například spjaté sezónní práce, uzavírání obchodů apod. Sezónní složku zobrazuje obrázek 2.3.

Obrázek 2.3: Sezónní složka

Cyklická složka C_t
Cyklická složka obsahuje cykly dlouhodobého hlediska oproti sezónní složce, kde časové období je například jeden rok. Cyklická složka většinou nemá konstantní periodu a může mít různou amplitudu. Cykly mohou být způsobeny různými faktory. Mohou to být ekonomické faktory, technologické, inovační, demografické aj. Cyklická složka zachytává kolísání okolo trendu, kdy se střídají fáze růstu a poklesu. Cyklickou složku představuje následující obrázek.

Obrázek 2.4: Cyklická složka

Příkladem cyklické složky je schodek rozpočtu ČR.

Náhodná složka E_t

Poslední složkou dekompozice řady je náhodná složka, také označována jako složka *reziduální*. Reziduální složka obsahuje náhodné výkyvy řady. Dále se může jednat o chybu měření, zaokrouhlovací chyby apod. Reziduální složka je zobrazena na obrázku 2.5.

Obrázek 2.5: Náhodná složka

Modely dekompozice časových řad

Pro dekompozici existují tři typy modelů dekompozice:

- Aditivní model
- Multiplikativní model
- Smíšený model

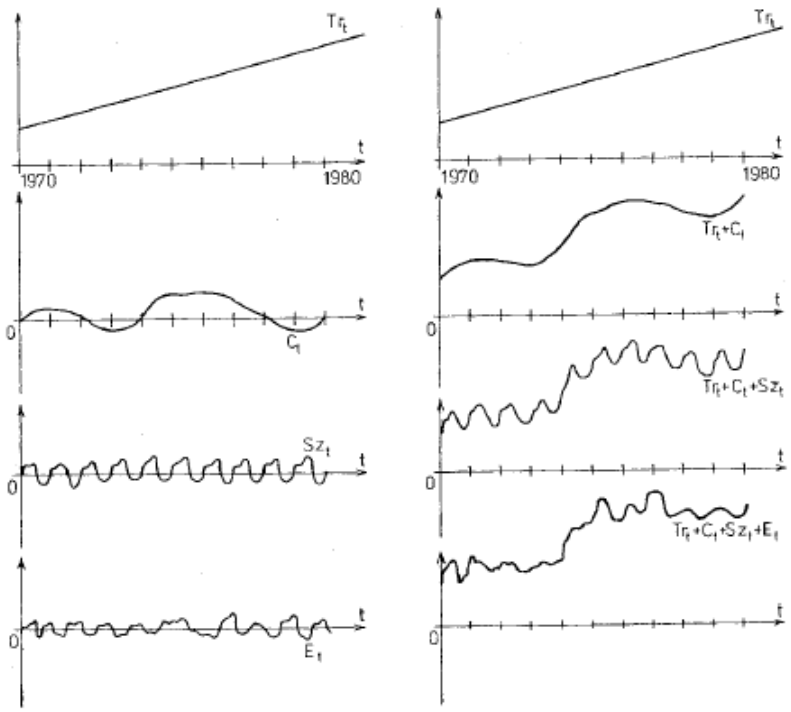
Vyjmenované modely určují, jak je výsledná časová řada složena z jednotlivých složek popsaných v předešlé části 2.10. [1]

Aditivní model

Výsledná časová řada aditivního modelu je součet všech jednotlivých složek. Složky

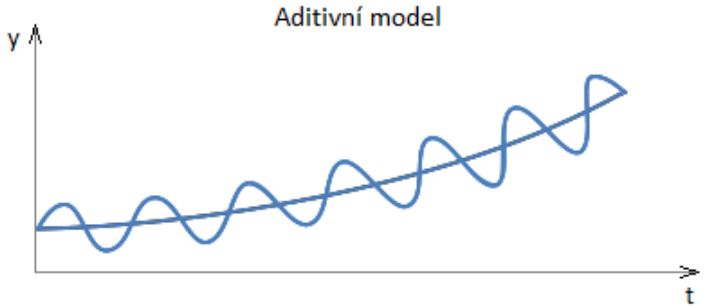
jsou uváděny v absolutní hodnotě a obsahují stejné jednotky. Aditivní model je dán rovnicí 2.17. [3]

$$Y_t = T_t + C_t + S_t + E_t \tag{2.17}$$



Obrázek 2.6: Znázornění skládání složek aditivního modelu (převzato z [4] obr. 7)

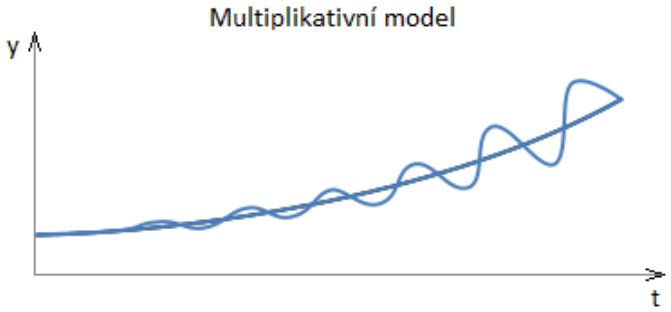
Aditivní model se využívá v případech, kdy rozptyl hodnot časové řady s časem neroste.



Obrázek 2.7: Aditivní model

Multiplikativní model
Výsledná řada tohoto modelu je oproti modelu aditivnímu složena ze součinů jednotlivých složek. Multiplikativní model se používá v případech, kdy rozptyl hodnot časové řady s časem roste. Jednotku obsahuje pouze složka *trend*, ostatní jsou považovány za bezrozměrné. Multiplikativní model je dán rovnicí 2.18. [3]

$$Y_t = T_t \cdot C_t \cdot S_t \cdot E_t \tag{2.18}$$



Obrázek 2.8: Multiplikativní model

Smíšený model
Jak již název napovídá, smíšený model je kombinací modelu aditivního spolu s modelem multiplikativním. Častým příkladem zápisu smíšeného modelu je rovnice 2.19. [1]

$$Y_t = T_t \cdot C_t \cdot S_t + E_t \tag{2.19}$$

Volba modelu závisí na zkoumané časové řadě, resp. na ukazateli, který řadu reprezentuje. V rozhodování nám může být nápomocný graf. V aditivním modelu sezónní složka osciluje kolem trendu s konstantní amplitudou, kdežto v multiplikativním modelu se sezónní složka mění.

2.11 Metody dekompozice časových řad

Pro samotnou dekompozici časových řad existuje několik metod, které se liší způsobem pohledu na časovou řadu tzv. v pohledu na vývoji sledovaného ukazetele. Cílem dekompozice je identifikace trendu a sezónní složky. Složky časové řady byly představeny v kapitole 2.10. Pro identifikaci jednotlivých složek se v praxi používají následující čtyři metody dekompozice [1]:

- naivní modely
- vyrovnání trendu matematickou křivkou

- vyrovnání klouzavým průměrem
- exponenciální vyrovnání

V následujícím textu bude popsán stručný princip jednotlivých metod pro analýzu časových řad.

Naivní modely

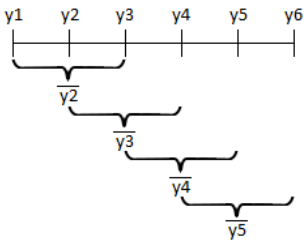
Naivní model je základní model dekompozice časové řady. Princip modelu spočívá ve vyjádření současné hodnoty sledované veličiny za pomocí hodnot minulých. Tento typ modelu se používá pro první iteraci hodnot před použitím přesnějších modelů.

Vyrovnání trendu matematickou křivkou

Tento typ modelu předpokládá neměnnost trendu v námi sledovaném úseku a lze ho tak popsat matematickou křivkou. Jedná se o tzv. *neadaptivní metodu*. Cílem modelu je určení matematické křivky spolu s odhadem jejích parametrů, tj. jde nám o nalezení předpisu funkce daného trendu.

Vyrovnání klouzavým průměrem

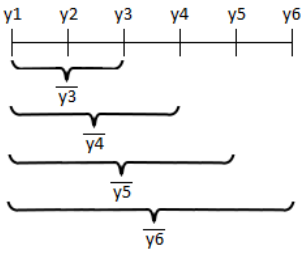
V některých případech nelze vyjádřit trend v námi zkoumané časové řadě jednou matematickou křivkou a to ani způsobem, který by rozdělil celý problém do menších částí, u kterých by se zjistili matematické funkce těchto částí a následně by se sjednotili v jeden celek. Způsob této *„techniky“* by mohl být velmi náročný, nebo by nemusel být vůbec realizovatelný a to z důvodu spojitosti zkoumaného trendu. Z tohoto důvodu se doporučuje využít *adaptivních metod*, do kterých patří *metoda klouzavých průměrů*. Metoda klouzavých průměrů je založena na průměrování hodnot intervalu řady nejčastěji o lichém počtu po sobě následujících hodnot časové řady, aby výsledná hodnota byla uprostřed zvoleného intervalu, tj. výsledný průměr hodnot je zpětně zapsán na index poloviny intervalu průměrované řady. Postup je znázorněn na obrázku 2.9.



Obrázek 2.9: Vyrovnání klouzavým průměrem

Exponenciální vyrovnání

Poslední představovanou metodou dekompozice časových řad je metoda exponenciálního vyrovnání trendu. Jedná se o další *adaptivní metodu*, která na rozdíl od metody *Vyrovnání klouzavým průměrem* vyrovnává všechny předešlé hodnoty a ne jen určitý interval. S rostoucím intervalem vyrovnání mají předešlé hodnoty exponenciálně klesající váhy. Způsob exponenciálního vyrovnání je znázorněn na obrázku 2.10.



Obrázek 2.10: Exponenciální vyrovnání

3 Data mining

V současné době se ve všech odvětvích společnosti elektronicky zaznamenává a uchovává obrovské množství dat na rozdíl od dřívějších dob, kdy data byla zaznamenává taktéž, ale ne v takovém množství. Velkým problémem byla hlavně forma, ve kterých byla data zaznamenána. Nejčastěji se data uchovávala v papírové podobě, tudíž analýza byla velmi zdlouhavá nebo dokonce nemožná. Data byla také velmi ovlivněná chybami vznikající při pořizování dat. V záznamech se často například objevovaly například duplicity, záznamy byly nekompletní či data obsahovala dokonce vymyšlená data. To sa stávalo, když někteří pořizovatelé neviděli smysl následného užití dat, uchovávalo se příliš velké množství dat, často duplicitně vymáháné řídícími orgány. Ještě v nedávné minulosti bylo z výše zmíněných důvodů pořizování dat různě bojkotované. Současná digitální doba se tyto vlivy snaží eliminovat a získávat relevantní data. Nastává otázka zda lze pořizovaná data smysluplně využít. Odpověď na tuto otázku dává ***data mining***, často označováno v českém překladu jako *dolování dat*. *Data mining* je určen k nalezení skrytých a užitečných informací ze zkoumaných dat. Díky této technice je možné z dat předvídat budoucí trendy, vývoj nemocí, předpověď prodejů nebo klasifikaci zákazníků. S analýzou dat mám zkušenosti z profesního života, jelikož pracuji jako konzultant/vývojář v oblasti SAP Business Warehouse a pomocí analýzy dat vytvářím datové výstupy nejčastěji ve formě reportů a to jak z oblasti výroby, prodejů, tak i z oblasti ***Customer Relationship Management*** známé častěji pod zkratkou ***(CRM)***. Data z oblastí jako je právě například výroba jsou často reprezentovány ve formě časových řad. V [17] je publikováno využití dat provozním centrem Rolls-Royce ve Velké Británii. Společnost nepřetržitě monitoruje přes 3700 prodaných leteckých motorů na celém světě, aby předvídala poruchy, které by mohly nastat po detekci signifikantních problémů. Společnost není jen výrobcem, ale také poskytovatelem této data miningové služby a podle [17] tvoří až 70% ročních tržeb divize pro výrobu dopravních letadel.

3.1 Příprava dat

Před nasazením data miningových nástrojů je důležitým krokem příprava samotných dat. Příprava dat je časově náročný a obtížný krok a zabere velkou část celého řešení. Cílem je ze *„surových“* dat vybrat potřebné údaje a upravit je do tvaru, který je vhodný pro další zpracování, např. pro různé algoritmy apod. Nejčastěji se data upravují do datové matice. Matice je strukturována do řádků a sloupců, kde řádky

označují záznamy a sloupce jejich atributy. Strukturou jsou tedy stejné jako databázová relace v relačních databázích. Zásadní rozdíl spočívá v tom, že vstupní matice pro modelování v data miningu vzniká opačným postupem než je tomu při návrhu datového modelu pro relační databázový systém. Návrh datového modelu v relačních databázích řeší redundanci dat dekompozicí dat do tabulek, které jsou provázány relacemi a spojeny pomocí klíčových atributů. Naopak data v data miningu jsou z různých zdrojů, nejen databázových, a spojují se do jedné datové matice. Matice nemusí být relací a tudíž může obsahovat duplicity a redundance, které když mají smysl, nevadí.

Příprava dat zahrnuje několik datových operací. Mezi ně například patří zpracování nekompletních záznamů (např. několik chybějících atributů v záznamu), odvozování atributů, redukci záznamů nebo atributů. Mezi operace, které se provádějí při chybějících hodnotách, patří odstranění nekompletních záznamů nebo doplnění chybějící hodnotou. Doplnění nekompletních hodnot záznamu může být například hodnotou *„null“*. Dále je možné nekompletní záznamy doplnit libovolnou hodnotou, průměrem (pokud se jedná o číselnou hodnotu) apod. Proces nahrazení chybějicích hodnot musí být uvážený a logicky spojený se zamýšleným použitím atributu v následném modelu. Některé další možnosti doplnění byly představeny v kapitole 2.7. Někdy je vhodné velké datové zdroje redukovat a vybrat pouze část dat, které budou dále vybrané. Redukcí počtu záznamů je možné zrychlit následnou analýzu dat. Mezi další redukce vybudované vstupní matice patří redukce atributů. Důvodem je například korelace atributů, významová redukce apod. Atributy datového souboru je také možno odvozovat. Odvozováním atributů se rozumí vytvoření nového atributu na základě jiného atributu, nebo většího počtu atributů. Příkladem může být odvození týdne z atributu obsahující datum.

Dalším zpracováním je úprava matice pro použité algoritmy ve vlastní analýze. Jak již bylo zmíněno v kapitole 2.8, některé algoritmy vyžadují numerické vstupy jiné pouze kategoriální data. Z tohoto důvodu musí být někdy hodnoty převedeny do numerické podoby a to například v případě, že algoritmus vyžaduje pouze kvantitativní (číselná) data. Typickým převodem je vytvoření substituce, tzv. kategoriální hodnoty nahradit stanoveným číslem. To platí pouze pro ordinální typ dat, tj. kategorie, kterou lze seřadit. U nominálních dat, tzv. neseřaditelná kategorie, se obvykle provádí převod na indikátorové proměnné. Naopak pokud s číselnými hodnotami algoritmus pracovat neumí, nebo vyžaduje kategorie, musí se data transponovat na intervalové kategorie. Tento proces může být relativně jednoduchý. A to když dělení do intervalů je dáno počtem vyskytujících se hodnot podle percentilů, nebo pevnou velikostí intervalu. Pokud do kategorizace promítneme vztah k predikované hodnotě, algoritmus návrhu intervalu pro optimální kategorizaci může být komplikovanější, je-li v modelu definovaná predikovaná proměnná.[2]

Nastíněné postupy mají pouze ilustrovat techniky práce s daty ve fázi přípravy datové matice. Porozumění datům, logickým vztahům a cílům data miningové úlohy teprve rozhoduje o použitých technikách a postupech. Obvykle je to náročný a ne-

triviální proces.

3.2 Data miningové nástroje

V dnešní době existuje velká řada data miningových nástrojů, které nabízejí řadu různých algoritmů a technik pro zkoumaní dat. Z tohoto důvodu bude představeno pouze několik vybraných data miningových programů. Hlavní pozornost je věnována dvěma nástrojům používaných v předmětu MTI/DM zejména z hlediska doporučení jejich použití pro typické problémy data miningových úloh.

Orange

Orange je open source nástroj, který je určen pro vizualizaci a analýzu dat. Nástroj je dostupný ve formě Python knihovny a skripty z této knihovny pak mohou být spuštěny v terminálu, nebo v prostředích jako je PyCharm apod. Orange je vyvíjen Bioinformatickou laboratoří na fakultě počítačových a informačních věd na Slovinské univerzitě Ljubljana, kde je dále s komunitou vyvíjen. Data mining se v tomto nástroji uskutečňuje pomocí vizuálního programování, nicméně lze také využít programovací jazyk Python. Orange obsahuje komponenty pro machine learning, nástroje pro bioinformatiku a text mining spolu s nástroji pro datovou analýzu. [5]

Weka

Weka obsahuje kolekci algoritmů pro ***machine learning***[1]. Tyto algoritmy mohou být přímo aplikovány na datové soubory pomocí vizuálního prostředí, nebo mohou být volány za pomocí vlastního programového kódu v jazyce *Java*. *Weka* disponuje nástroji pro již zmíněný machine learning, data mining, klasifikaci, regresi, asociační pravidla a další. Nástroj *Weka* je vyvíjen univerzitou Waikato na Novém Zélandě. Předpokladem nástroje je, že data jsou uložena v jednom souboru, či jedné tabulce. Další možností je využít nástroj pro dotazování se nad databází a ke zpracování obdržených výsledků. [6]

Rattle GUI

Jedná se o grafické uživatelské rozhraní, které poskytuje data miningové nástroje používající se skrze programovací jazyk *R*. Výstupy nástroje Rattle jsou reprodukovány v grafické podobě. [7]

RapidMiner

RapidMiner je prostředí poskytující nástroje pro machine learning, data mining a prediktivní analýzu. Nástroj umožňuje vytvářet analytický workflow, tzv. vytvářet

[2]Věda zabývající se algoritmy algoritmy umělé inteligence, které jsou schopné se samy přizpůsobovat a učit se. [2]

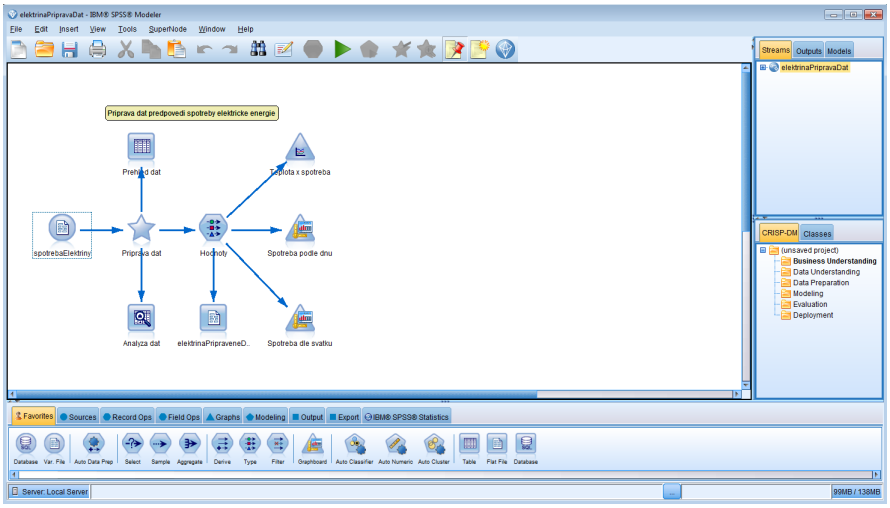
grafické schéma, kde jednotlivé uzly představují algoritmy a další operace. Dále *RapidMiner* umožňuje používat algoritmy z jiných nástrojů a to z nástrojů *Weka* a *R*. Výhodou je možnost rozšířit *RapidMiner* dalšími funkcemi za pomocí volně dostupných rozšířeních. [8]

Mezi další nástroje patří nástroj ***IBM SPSS Modeler*** a nástroj ***Knime***. *IBM SPSS Modeler* a *Knime* byly vybrány na základě používání ve výuce a proto v nich jsou případové studie modelovány.

IBM SPSS Modeler

Jedná se o datamingový nástroj, který slouží pro zpracování celého data miningového procesu. Data miningový proces zahrnuje získání dat i uvedení výsledků do reálného nasazení. Nástroj je vyvíjen společností ***IBM***, která se již desítky let zabývá informačními technologiemi.

IBM SPSS Modeler nabízí celou řadu vestavěných funkcí pro zpracování dat a také mnoho modelovacích algoritmů. Nástroj poskytuje podporu jazyka *R* a je do něj možné integrovat nástroje využívající tento jazyk. *IBM SPSS Modeler* podporuje integraci softwaru *SAP HANA*, *Oracle R* a další databázové stroje založené na jazyku *R*. Všechny obsažené funkce data miningového nástroje a algoritmy jsou uspořádány do přehledného editoru, ve kterém celý proces zpracování a modelování dat probíhá v grafické podobě, to znamená, že celé modelování je tvořeno jako grafické schéma. Ve schématu jsou jednotlivé uzly, které reprezentují určitý algoritmus či jiné datové operace, propojovány do orientovaného grafu. V orientovaném grafu jsou uzly mezi sebou propojeny a to od uzlů, které reprezentují zdroj dat až po námi definovaný výstup. Seskupení uzlů se označuje jako proud neboli ***stream***. Na následujícím obrázku je znázorněn příklad datového proudu, který je realizován v *IBM SPSS Modeleru*.[9]



Obrázek 3.1: Data miningové prostředí IBM SPSS Modeler

Licence

IBM SPSS Modeler je nabízen v několika verzích, které se liší v nabízených funkcí:

SPSS Modeler Personal
Jedná se o základní verzi tohoto software. Verze *Personal* je určena pouze pro osobní počítače, tudíž není možné využívat ***Analytic server*** používající se pro zpracování ***Big Data***. Cena základní licence činí 4 530$ na rok pro jednoho uživatele a nabízí roční technickou podporu.

SPSS Modeler Professional
Oproti minulé verzi verze *Professional* nabízí pokročilejší algoritmy a techniky pro přípravu dat. Verze *Professional* navíc disponuje možností provádět výpočty na již zmíněném *Analytic serveru*. Cena licence je 6 800$ pro uživatele na jeden rok.

SPSS Modeler Premium
Verze *Premium* nabízí jak výpočet na uživatelském počítači, tak i možnost přenést výpočet na *Analytic server*. Dále nabízí možnost analýzy sociálních sítí, nebo *Text mining*. Cena licence je 11 300$ pro jednoho uživatele na jeden rok.

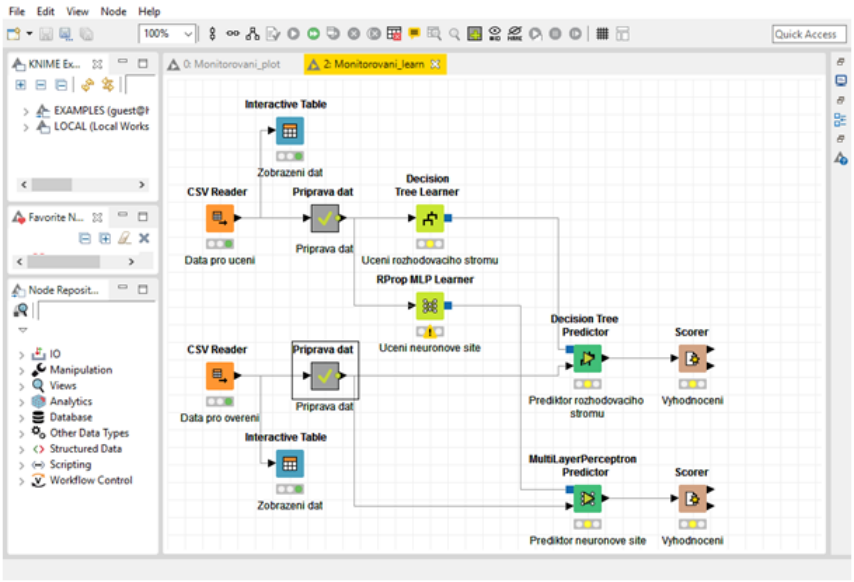
Veškeré ceny licencí jsou aktuální ke dni 1.5.2016 [10]

Knime

Jedná se v základní verzi o volně dostupný software pro analýzu dat. Platforma nabízí mnoho různých komponent (uzlů) pro *Machine learning* a data mining. *Knime* aktivně vyvíjí skupina Michaela Bertholda na univerzitě Konstanz v Zurichu. Nástroj je vyvíjen od roku 2006, kdy byl prvně využíván pro farmaceutické výzkumy. Později byl *Knime* použit v ostatních odvětvích jako business inteligence, nebo analýza finančních dat. Velkým rozdílem licenční politiky platformy *Knime* je v poskytování uzlů a algoritmů. Všechny nabízené licence obsahují stejné uzly a lze tedy stejný datový tok vytvořit v jakékoliv licenci na rozdíl od nástroje *IBM SPSS Modeler*, který ve vyšších verzí poskytuje uzly neobsahující v základní verzi. Platforma ve vyšších licencí nabízí řadu doplňkových služeb, mezi které patří například vzdálené spouštění datových toků, plánované spouštění, autentifikaci uživatele pro přístup k datovým tokům umístěných na serveru a také webové API pro konfiguraci webových služeb. Cena nástroje *Knime* ve své nejvyšší licenci, která je označována jako *Knime server* poskytující veškeré funkce platformy, činní 21 000 pro 5 uživatelů na jeden rok. [20]

Stejně jako nástroj *IBM SPSS Modeler*, *Knime* je koncipován jako editor, ve kterém se zpracování a úprava dat provádí pomocí k tomu určených uzlů. Uzly tvoří datový proud nebo-li ***workflow***. Výhodou nástroje je možnost vyvíjet své vlastní moduly, nebo používat uzly, ve kterých lze požadované chování naprogramovat a to za pomocí programovacího jazyka *JAVA* a *R*. Další výhodou jsou dostupné příklady. Příklady jsou zaměřeny na konkrétní problémy a je pro ně vždy vytvořena

samostatná sekce, ve které může uživatel nalézt různé varianty řešení. [11] Přehled *workflow* v tomto prostředí lze vidět na následujícím obrázku.



Obrázek 3.2: Data miningové prostředí Knime

4 Případové studie

V diplomové práci byly vypracované dvě případové studie v data miningovém nástroji *Knime* a *IBM SPSS Modeler* a to především proto, aby byla ověřena možnost práce s časovými řadami ve vybraných, spíše technických než ekonomických úlohách. První studie se nazývá monitorování provozu strojů a je určena na sestavení modelu, který je schopen predikovat selhání stroje. Vychází se historie, kde je *„mapován“* provoz stroje až do jeho zastavení při selhání sledováním veličiny výkonu stroje, teploty stroje, tlaku a dalších parametrů. Pomocí těchto parametrů se model *„naučí“* rozeznávat z provozu stroje chybové stavy za pomocí naměřených časových řad. Druhá studie je zaměřena na predikci spotřeby elektrické energie. Cílem druhé studie je vytvořit model, který z naměřených časových řad (teploty, spotřeby, ...) bude schopen predikovat spotřebu el. energie. Datové soubory obsahují jak data pro učení, tak i pro testování. Za tímto účelem byla použita data, která mi byla poskytnuta a budou použita ve výuce v předmětu data miningu.

Případové studie jsou modelovány v data miningových nástrojích *Knime* a *IBM SPSS Modeler*. Cílem celé práce bylo porovnat volně dostupný nástroj *Knime* a jeho aktuální nástroje pro přípravu dat, vlastní modelování, testování a případné nasazení do provozu s profesionálním nástrojem *IBM SPSS Modeler*. V následujícím textu jsou popsány obecné kroky provedené v každé studii a budou zde popsány pouze rozdíly implementace ve zvolených data miningových nástrojích. Konkrétní implementace modelů ve vybraných nástrojích je popsána na přiloženém CD a bude k dispozici studentům při studiu časových řad v předmětu data mining na fakultě mechatroniky.

Vlastní struktura případových studií je popsána následujícími body:

- cíl studie
- struktura datového souboru
- analýza závislostí dat
- vytvoření modelu
- testování modelu na testovacích datech a jeho vyhodnocení

4.1 Algoritmy případových studií

Predikce modelů případových studií jsou založeny na modelování pomocí rozhodovacích stromů a neuronových sítí. Důvod výběru algoritmu rozhodovacích stromů a neuronové sítě vyplynul ze zadání úloh. Případové studie jsou klasifikačního typu a tudíž zmíněné algoritmy jsou pro tento typ úloh vhodné.

Při tvorbě případových studií v nástrojích *IBM SPSS Modeler* a *KNIME* byla snaha dodržet shodnou strukturu datových toků a vybrat stejné rozhodovací algoritmy. Bohužel oba nástroje nedisponovaly shodnými algoritmy a tak byly vybrány takové algoritmy, které se spolu nejvíce podobají. V nástroji *IBM SPSS Modeler* byl vybrán rozhodovací strom s algoritmem ***C5.0***, který se pro takovéto typy úloh často využívá. Prostředí *Knime* niceméně tímto algoritmem nedisponuje, ale nabízí rozhodovací strom, který obsahuje modifikaci algoritmu **C4.5**. [**?**]. Pro neuronovou síť byl v prostředí *Knime* vybrán uzel ***RProp MLP*** což značí neuronovou síť typu resilent backpropagation. *Knime* dále nabízí neuronovou síť typu ***PNN (Probabilistic neural network)***, nicméně tato neuronová síť při testování vykazovala úspěšnost v modelu *monitorování provozu strojů* pouhých 42%. Další typy neuronových sítí prostředí *Knime* v testované verzi nenabízí. V nástroji *IBM SPSS Modeler* je k dispozici pouze jeden uzel neuronové sítě, který byl nastaven jako ***MLP*** neuronová síť. Neuronová síť typu *MLP* disponuje výběrem počtu neuronů ve skrytých vrstvách. Neuronová síť v nástroji *IBM SPSS Modeler* dále nabízí možnost časového omezení prováděného výpočtu, nebo je možné nastavení počáteční inicializační hodnoty pro získání stejných výsledků při každém dalším spuštění modelu.

4.1.1 Klasifikační algoritmy

Rozhodovací strom je jeden z klasifikačních algoritmů, který je ve výsledku složen z uzlů a listů, kde uzly testují hodnotu atributu a listy představují výsledky těchto testů tzv. obsahují výsledky klasifikace. Klasifikační stromy pracují pouze s kategoriálními daty a proto je nutné data do tohoto typu převést. Některé algoritmy provádějí konverzi automaticky. Rozhodovací stromy obsahují několik klasifikačních algoritmů. Mezi klasifikační algoritmy patří *ID3, C4.5, C5.0, SPRINT* a další. V případových studiích se využívá algoritmus *C4.5* a *C5.0*. Algoritmus *C4.5* je založen na algoritmu ***ID3***. Algoritmus *ID3* pracuje na principu klasického rozhodovacího stromu, který byl popsán výše a tedy je založen na testování hodnoty vstupního atributu v uzlu a přiřazení třídy v listech stromu. [14]

Algoritmus buduje strom hledáním nejlepšího prediktora (atribut) pro predikovanou proměnnou pomocí informačního zisku, který nejlépe buduje další úroveň rozhodování na trénovacích datech tzv. má nejvyšší ***informační zisk***. *Informačním ziskem* (*Gain(A)*) se rozumí snížení entropie při znalosti atributu A, tedy rozdíl entropie pro cílový atribut (celá data) a ***entropie*** uvažované třídy atributu. Cílem volby vhodného atributu v rozhodovacím stromu je co nejlepší odlišení vstupních dat vzhledem k cílovému atributu (predikované proměnné).

Entropie

Systém S má T diskrétních navzájem se vylučujících stavů s_i kde $i = 1 \dots T$, p_i jsou relativní četnosti výskytu stavu s_i a suma p_i je rovná jedné, potom

$$H(S) = -\sum_{i=1}^{T}(p_t \cdot log_2 p_t) \tag{4.1}$$

definuje entropii, která vyjadřuje míru neuspořádanosti systému S.

Každý kategoriální atribut A lze chápat jako systém ve výše uvedeném smyslu, jeho jednotlivé kategorie jako stavy s_i a relativní četnosti p_i lze snadno spočítat z trénovacích dat. Pro hledání vhodného atributu pro větvení je používaná entropie možných prediktorů podmíněná cílovým atributem C tj. predikovaným atributem $H(\frac{A}{C})$. Volí se ten atribut, který má nejmenší entropii.

Výpočet informačního zisku

Informační zisk je míra odvozená od entropie a měří redukci entropie při volbě prediktora A. Tedy o kolik se sníží celková entropie cílového atributu C volbou atributu A. Počítá se podle vztahu

$$Gain(A) = H(C) – H(\frac{A}{C}) \tag{4.2}$$

Informační zisk pro hledání vhodného prediktora nad danou množinou dat se spočítá pro všechny možné prediktory a volí se ten, kde informační zisk je největší. Pro výběr prediktorů při budování rozhodovacího či klasifikačního stromu se používají i jiné statistiky či algoritmické postupy. Volba, těch co zde popisuji, je důsledkem použitých nástrojů v případových studiích.

Informační zisk se využívá v následujících algoritmech:
Algoritmus *C4.5* je modifikace algoritmu *ID3*, který je vylepšen o přidání prořezávání, umožňuje zpracovávat spojitá data a to tím, že je automaticky diskretizuje a také je schopen zpracovávat chybějící hodnoty, které se nezahrnují ve výpočtech entropie a informačního zisku.

Následníkem algoritmu *C4.5* je komerční verze algoritmu *C5.0*, který je vylepšen o nižší nároky na paměť, dosahuje vyšší rychlosti klasifikace, umožňuje boosting a další.

Informační zisk je možné modifikovat na tzv. ***poměrný informační zisk (information gain ratio)***. *Poměrný informační zisk* bere v potaz i počet hodnot atributu, rozuměj počet kategorií, tedy když by byla stejná hodnota pro atribut v celé datové množině, tj. existovala by jediná kategorie pro danou datovou množinu,

$p_i = 1$ pro $i = 1$ a výsledná entropie by byla nulová. Takovýto atribut je nepoužitelný pro klasifikaci a je automaticky z klasifikace vyloučen. [15]

Rozhodovací stromy obsahují další typ klasifikátoru dat, který hledá vhodný prediktor na základě ***Gini indexu***. ***Gini index*** využívá například algoritmus *CART, SPRINT* a další.
Gini index se spočítá následujícím vzorcem:

$$GINI(A) = 1 - \sum_{t=1}^{T} p_t^2 \tag{4.3}$$

kde p_t označuje relativní četnost pro kategorii t atributu A.

Pokud hodnota *Gini indexu* nabývá hodnoty nula, pak je v konečném uzlu jediná kategorie. Pokud *Gini index* nabývá hodnoty 1, je v každé kategorii stejný počet hodnot. [16]

Nastavení rozhodovacích stromů v použitých nástrojích

Rozhodovací stromy obsahují řadu nastavení. Následující text popisuje možnosti nastavení rozhodovacích stromů v nástrojích *IBM SPSS Modeler* a *Knime*.

IBM SPSS Modeler - uzel C5.0

- **boosting** - provede se standardní klasifikace a na špatně klasifikované výsledky se znovu aplikuje nový model a tak dále. Počet modelů je omezen uživatelem - výsledkem je zvýšení přesnosti vyhodnocování rozhodovacího stromu
- **pruning severity** - určuje míru prořezávání, lze nastavit číslo v rozmezí 0 až 100. Čím blíže se číslo blíží hodnotě 0, tím přesnější je výsledný strom tzv. obsahuje více informací a opačně.
- **Min number records per node** - počet záznamů na uzel, používá se jako stop kritérium algoritmu

Nastavení uzlu vychází z dostupné dokumentace *IBM SPSS Modeler* viz. [12].

Knime - uzel Decision Tree Learner

- **Quality measure** - klasifikační kritérium stromu - *Gini index* a *Gain ratio*
- **Pruning method** - slouží k prořezávání stromu a předcházení tzv. overfiting což značí, že model se učí i šumy, které se v datech mohou vyskytovat, a tím se může snižovat přesnost modelu na testovacích datech.
- **Number threads** - počet vláken procesu pro výpočet

Nastavení uzlu vychází z dostupné dokumentace *Knime* viz. [13].

4.2 Monitorování provozu strojů

Studie monitorování zkušebního provozu je zaměřena na predikci selhání stroje periodickým sledováním jeho parametrů jež jsou tlak stroje, teplota stroje, výkon stroje, aktuální stav stroje, stav stroje na konci časové řady a čas od poslední kontroly. V této úloze bude vytvořen model, který ze vstupních dat, tj. trénovací množiny ve formě časové řady, se naučí předpovídat selhání stroje. Vytvořený model bude aplikován na testovací data a následně bude vyhodnocena jeho přesnost. Pro učení byly zvoleny rozhodovací stromy a neuronová síť.

4.2.1 Struktura datového souboru

Monitorování zkušebního provozu obsahuje dva datové soubory. První z nich je datový soubor pro *„učení“* modelu a je označen jako ***monitorovaniUceni***, neboli trénovací množina. Druhý datový soubor obsahuje testovací data a nese označení ***monitorovaniTestovani***. Testovací data slouží pro zjištění přesnosti modelu. Přehled záznamů datového souboru je zobrazen na obrázku 4.1.

Row ID	Time	Power	Temper...	Pressure	Uptime	Status	Outcome
Row77	77	902	259	0	404	0	303
Row78	78	894	259	0	404	0	303
Row79	79	888	259	0	404	0	303
Row80	80	884	259	0	404	0	303
Row81	81	883	259	0	404	0	303
Row82	82	882	259	0	404	0	303
Row83	83	880	259	0	404	0	303
Row84	84	873	259	0	404	0	303
Row85	85	866	259	0	404	0	303
Row86	86	858	259	0	404	0	303
Row87	87	850	259	0	404	0	303
Row88	88	844	259	0	404	0	303
Row89	89	839	259	0	404	0	303
Row90	90	834	259	0	404	303	303

Obrázek 4.1: Přehled struktury záznamů datového souboru v monitorování zkušebního provozu

V následující tabulce jsou popsány jednotlivé atributy datových souborů.

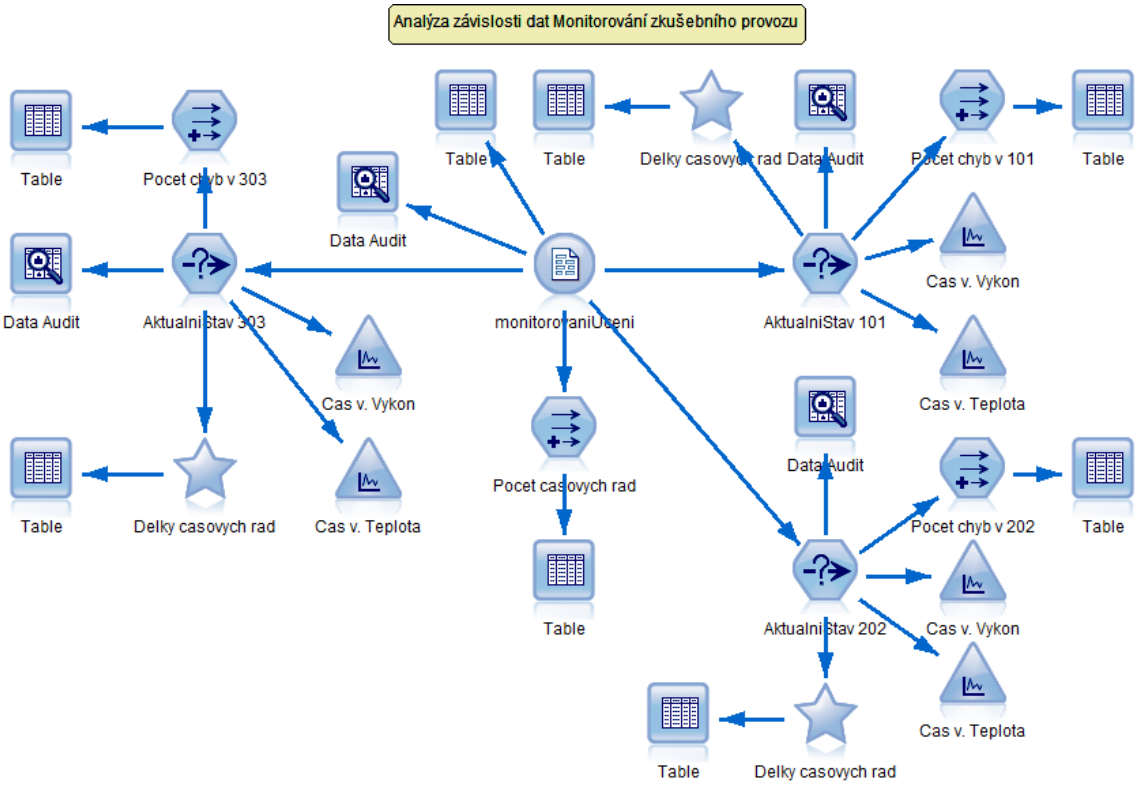
Tabulka 4.1: Atributy datového souboru - Monitorování zkušebního provozu

Název	***Akce***
Cas	čas pořízeného záznamu (pořadí záznamu)
Vykon	výkon stroje [W]
Teplota	teplota stroje [°C]
Tlak	tlak stroje, hod. 0 - normální stav, hod. 1 - výstr. stav
PosledniKontrola	čas od provedené poslední kontroly
AktualniStav	stav stroje, 0 normální stav (101, 202 a 303 jsou chyb. stavy)
KonecovyStav	chybový stav, do kterého se dostane stroj na konci čas. řady

Dalším krokem studie je nalezení závislostí mezi atributy datového souboru a následné vytvoření modelu pro předpověď poruchy strojů.

4.2.2 Analýza vstupních dat

Pro zjištění závislostí mezi atributy byl zvolen datový soubor *monitorovaniUceni*. Vstupní data jsem podrobil analýze pro porozumění dat před samotnou tvorbou modelu pro předpověď poruchy strojů. Před započetím manipulaci s daty jsem si pomocí uzlu ***Table*** zobrazil vstupní matici pro přehled vstupních dat. Velmi nápomocný pro rychlou analýzu vstupu je uzel ***Data audit***, který slouží k provedení rychlé analýzy bez nutné manipulace s daty. Výstupem uzlu je přehled atributů vstupní matice spolu se zobrazením maximálních, resp. minimálních hodnot číselných atributů, průměrná hodnota a také celkový počet záznamů, počet zastoupení kategorií atributu apod. Zde je nutné zmínit fakt, že uzel *Data audit* se nachází pouze v nástroji *IBM SPSS Modeler* a konkurenční nástroj *Knime* obdobnou variantou bohužel nedisponuje. Cílem analýzy závislosti atributů je zjistit chování stroje v chybových stavech a odhalit atributy, které nejvíce ovlivňují stroj. Chybovým stavem je v datovém zdroji označen atribut *AktualniStav* a může nabývat číselných hodnot 101, 202 a 303. Za tímto účelem jsem separoval záznamy s těmito hodnotami do třech skupin. V každé skupině jsou pak analyzovány závislosti atributů. Ve studii je v jednotlivých chybových stavech monitorována s přibývajícím časem teplota a výkon. Struktura uzlů pro nalezení závislostí atributů zachycuje obrázek 4.2.



Obrázek 4.2: Analýza závislosti atributů v monitorování zkušebního provozu v *IBM SPSS Modeler*

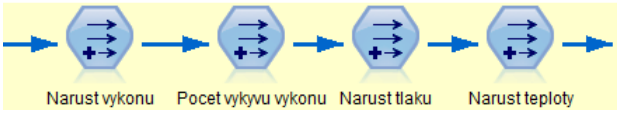
Výsledkem analýzy závislostí je, že při chybovém stavu 101 je teplota s časem konstantní, ale výkon klesá. Chybový stav 202 indikuje, že teplota stroje s časem roste a výkon mírně osciluje. Při chybovém stavu 303 je naopak teplota stroje konstantní a výkon stroje klesá. Atributy teplota a výkon stroje jsou tedy důležitými atributy pro predikční model. Další uzly analýzy dat jsou uzly pro zjištění délky jednotlivých časových řad, zjištění počtu chyb apod. Samozřejmě by nás dále mohly zajímat maximální, resp. minimální hodnoty tlaku, teploty a další informace. Na přiloženém CD v souboru (*Monitorovani provozu stroju - zkoumani dat*) je k dispozici několik otázek, kterými jsem analyzoval datový soubor použitý v tomto modelu a následně to vedlo k porozumění datům.

4.2.3 Příprava dat

Při tvorbě data mingového modelu je v prvním kroku potřeba připravit data před dalším použitím. Příprava dat je důležitým krokem před tvorbou modelů či dalším zkoumáním. Několik metod přípravy dat bylo popsáno v kapitole 3.1.

V přípravě dat nás budou zajímat atributy měnícího se tlaku, teploty a výkonu. Vhodnými statistickými úpravami budeme chtít data upravit do takové podoby,

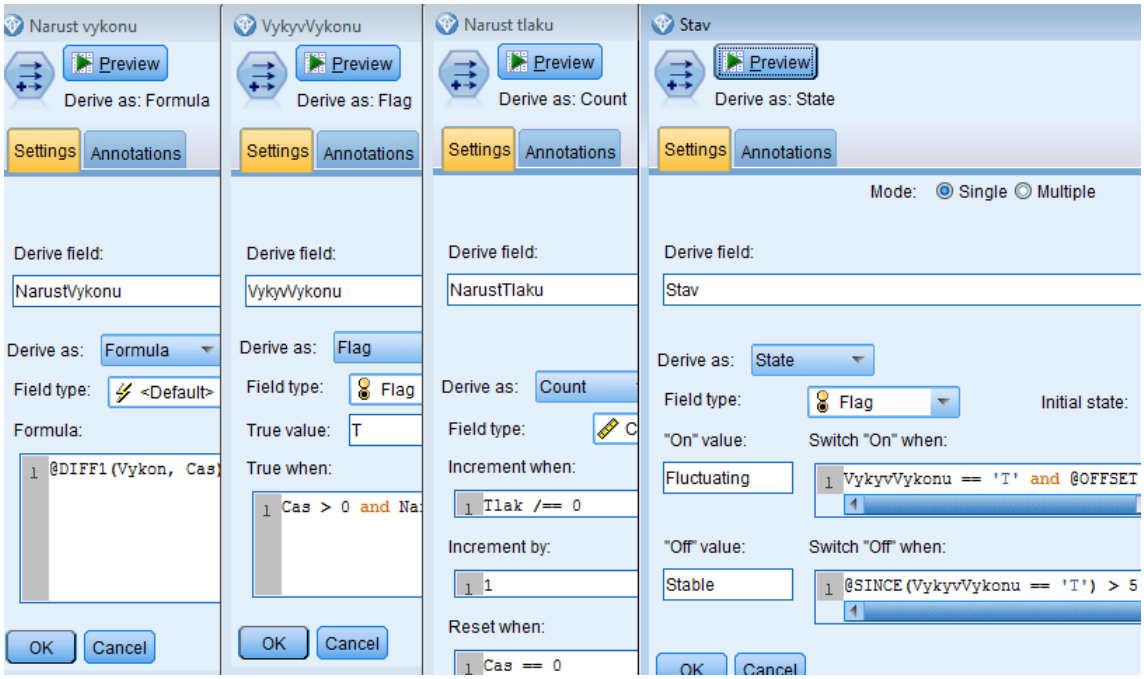
abychom zjistili určité vlivy mezi daty, které nastávají během provozu monitorovaného stroje. Cílem bude zjistit změnu (rozdíl) následujících hodnot časové řady, které budou vstupy pro další uzly využívající klouzavého průměru. Z tohoto důvodu bude příprava dat obsahovat několik nových uzlů, které budou popisovat změny zmíněných veličin tlaku, teploty a výkonu. V několika následujících bodech jsou popsány nově odvozené uzly.



Obrázek 4.3: Uzly využívající veličiny tlaku, teploty a výkonu v nástroji *IBM SPSS Modeler*

- **Narust vykonu** - slouží k výpočtu rozdílu výkonu ve dvou následujících záznamech
- **Narust teploty** – slouží k výpočtu rozdílu teploty ve dvou následujících záznamech

pro výpočet změny dvou následujících hodnot byla v nástroji *IBM SPSS Modeler* vybrána funkce ***@DIFF1(FIELD1, FIELD2)***, která vrací rozdíl prvního argumentu v závislosti na změně druhého argumentu. V našem případě to byly diference neboli přírůstek v časové řadě pro výkon stroje a v časové řadě monitorující teplotu. Odvození této nové proměnné vyšlo ze zkoumání časových řad ve fáze analyzování dat, které byly k dispozici. Výkon i teplota byly rostoucí či klesající, případně oscilující. Odlišit tyto stavy diferencí po sobě jdoucích hodnot je první fází přípravy dat. V nástroji *IBM SPSS Modeler* se toto chování realizuje pomocí uzlu ***Derive*** a správně zvolených parametrů. Uzel *Derive* poskytuje několik možností nastavení. Nastavení závisí na posouzení dataminera a konkrétní situaci. Několik variant nastavení zachycuje obrázek 4.4.



Obrázek 4.4: Nastavení uzlu *Derive* v nástroji *IBM SPSS Modeler*

Políčko *Derive field* v uzlu *Derive* je určené k pojmenování odvozeného atributu. Polem *Derive as* se nastavuje chování uzlu, které může být například operace *Count*. Zde se nastavuje počáteční podmínka, hodnota pro inkrementaci a podmínka *Reset when* určující nové podmíněné načítání.

Knime nicméně žádným obdobným uzlem pro poskytnutí takovýchto funkcí nedisponuje a tudíž bylo nutné toto chování naprogramovat. Platforma disponuje uzly, do kterých je možné psát programový kód v jazyce JAVA a je tedy možné *„ručně“* manipulovat s daty pomocí programovacího jazyka. Ukázka programového kódu jazyka JAVA v programovatelném uzlu *Java Snippet* nástroje *Knime* je zobrazen na obrázku 4.5.

```
14 // system variables
26 // Your custom variables:
27 int vykonPredesly = 0;
28 int casPredesly = 0;
29 // expression start
31 // Enter your code here:
32
33 if(c_Cas == 0)
34 {
35     out_NarustVykonu = (double)(c_Vykon - vykonPredesly)/(0-casPredesly);
36     casPredesly = c_Cas;
37     vykonPredesly = c_Vykon;
38 }
39 else
40 {
41     out_NarustVykonu = (double)c_Vykon - vykonPredesly;
42     casPredesly = c_Cas;
43     vykonPredesly = c_Vykon;
44 }
```

Obrázek 4.5: Ukázka prog. kódu JAVA v nástroji *Knime*

Další mapování vývoje atributů *výkon* a *tlak* jsou určeny následující uzly:

- **Pocet vykyvu vykonu** – vrací hodnotu TRUE pokud jsou dva následující záznamy výkonu opačné (liší se znaménka). Osciluje-li výkon je to charakteristické pro chybový stav.
- **Narust tlaku** – počet výstražných tlaků pro každou časovou řadu zvlášť

Pro nárůst výstražných tlaků se ukládá jejich počet v časové řadě do aktuálního záznamu. Situace je zachycena na obrázku 4.6

	Cas	Vykon	Teplota	Tlak	NarustTlaku	PosledniKontrola
734	65	895	252	0	0	284
735	66	895	252	0	0	284
736	67	892	252	0	0	284
737	68	888	252	1	1	284
738	69	887	252	0	1	284
739	70	887	252	1	2	284
740	71	887	252	0	2	284
741	72	885	252	0	2	284
742	73	885	252	0	2	284
743	74	885	252	0	2	284
744	75	882	252	0	2	284
745	76	878	252	0	2	284
746	77	877	252	0	2	284
747	78	873	252	0	2	284
748	79	870	252	0	2	284
749	80	870	252	0	2	284
750	81	870	252	0	2	284
751	82	868	252	1	3	284
752	83	867	252	1	4	284

Obrázek 4.6: Nárůst výstražných tlaků

Při stavu 101 dochází k prvnímu tlakovému výkyvu, který se opakuje nepravidelně až do selhání stroje.

Díky provedené analýze vstupních dat, ve které byla zjištěna závislost změny teploty a změny výkonu v čase v závislosti na chybovém stavu, jsem vytvořil dva nové uzly pro odvození nových atributů. Odvozovanými atributy jsou ***zmena teploty*** a ***zmena vykonu***. Uzly používají metodu klouzavého průměru pro vyhlazení možných výkyvů hodnot způsobené chybou měření či anomáliích, které mohly nastat. Velikost nastaveného klouzavého průměru ovlivňuje přesnost výsledného modelu a z tohoto důvodu byl vhodný klouzavý průměr vybrán na základě několika provedených měření, které jsou zaznamenány v následující tabulce. Měření bylo provedeno na již hotovém modelu a slouží pro zpětné zpřesnění modelu. Ve výchozím nastavení byl klouzavý průměr nastaven na hodnotu 5.

Tabulka 4.2: Závislost počtu záznamů klouzavého průměru na přesnost modelu

	Úspěšnost [%]	
Počet záznamů pro klouzavý průměr	**Strom**	**Neuronová síť**
1 záznam	91,22	95,34
3 záznamy	98,88	98,57
5 záznamů	99,68	99,35
7 záznamů	99,88	99,56
9 záznamů	99,93	99,56
11 záznamů	99,98	99,71

Provedené měření ukazuje závislost přesnosti modelu na zvoleném klouzavém průměru. Z uvedené tabulky 4.2 je vhodné nastavení klouzavého průměru na hodnotu 11 z důvodu největší přesnosti modelu. Hodnoty tabulky byly získány z nástroje *IBM SPSS Modeler* s nastavenou inicializační hodnotou 284 681 600 pro neuronovou síť.

Pro uzel výpočet nárůstu tlaku nebyla již využita stejná statistická metoda klouzavého průměru a to z důvodu, že tlak je reprezentován hodnotou 1 a 0 a nechceme vyhlazovat data, chceme znát výkyv hodnot, proto s tlakem pracujeme jinak. V dalším kroku byly vytvořeny následující uzly:

- **Zmena vykonu** - klouzavý průměr hodnot, které produkuje uzel *Narust vykonu*
- **Stav** – pokud jsou dva záznamy po sobě z uzlu *Pocet vyskytu vykonu TRUE*, uzel vrací hodnotu *fluctuating* (nestabilní), jinak vrací hodnotu *stable*. Z nestabilního stavu se vrací, pokud je po sobě pět záznamů (výchozí nastavení) z uzlu *Pocet vyskytu vykonu FALSE*.
- **Zmena teploty** – klouzavý průměr hodnot, které produkuje uzel *Narust teploty*

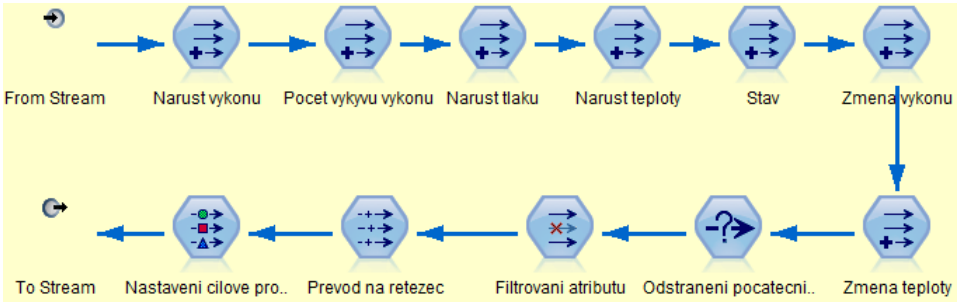
Stejně tak jako volba klouzavého průměru, nastavení hodnoty v uzlu *Stav*, po které uzel přechází zpět do stabilního stavu, ovlivňuje přesnost modelu. V následující tabulce je zobrazeno testování počtu záznamů na přesnost modelu při návratu do stabilního stavu.

Tabulka 4.3: Závislost počtu záznamů na přesnost modelu při návratu do stabilního stavu

Počet záz. pro stabilní stav	**Úspěšnost [%]**	
	Strom	**Neuronová síť**
1 záznam	98,88	98,88
3 záznamy	99,01	98,52
5 záznamů	99,68	99,35
7 záznamů	99,83	99,79
9 záznamů	99,88	99,83
11 záznamů	99,88	99,83

Hodnoty se ustálily na devíti záznamech s přesností modelu 99,88% pro rozhodovací strom a 99,83% pro neuronovou síť, když uzly *zmena teploty* a uzel *zmena vykonu* měli nastavený klouzavý průměru na hodnotu 5. Hodnoty byly naměřeny v nástroji *IBM SPSS Modeler*.

Zbývající část přípravy dat obsahuje několik dalších vytvořených uzlů. Jedním z nich je uzel pro filtrování dále nepotřebných atributů z datové matice, protože jsou zakompovány do jiných uzlů a pro další zpracování již nejsou potřeba. Mezi tyto atributy patří *Pocet vyskytu vykonu*, který je využit v uzlu *Stav*, čas, výkon, teplota a tlak a také nárůst teploty, nárůst výkonu a počet výkyvů výkonu. Dále je zapotřebí odstranit počáteční záznamy v každé časové řadě, protože v těchto záznamech nejsou definovány referenční charakteristiky a obsahují hodnoty *NULL*. Pro tyto operace jsou v každé platformě určeny jiné uzly a z tohoto důvodu je konkrétní implementace popsána v přílohách pro každou studii zvlášť. Obrázek 4.7 představuje přípravu dat v prostředí *IBM SPSS Modeler*.

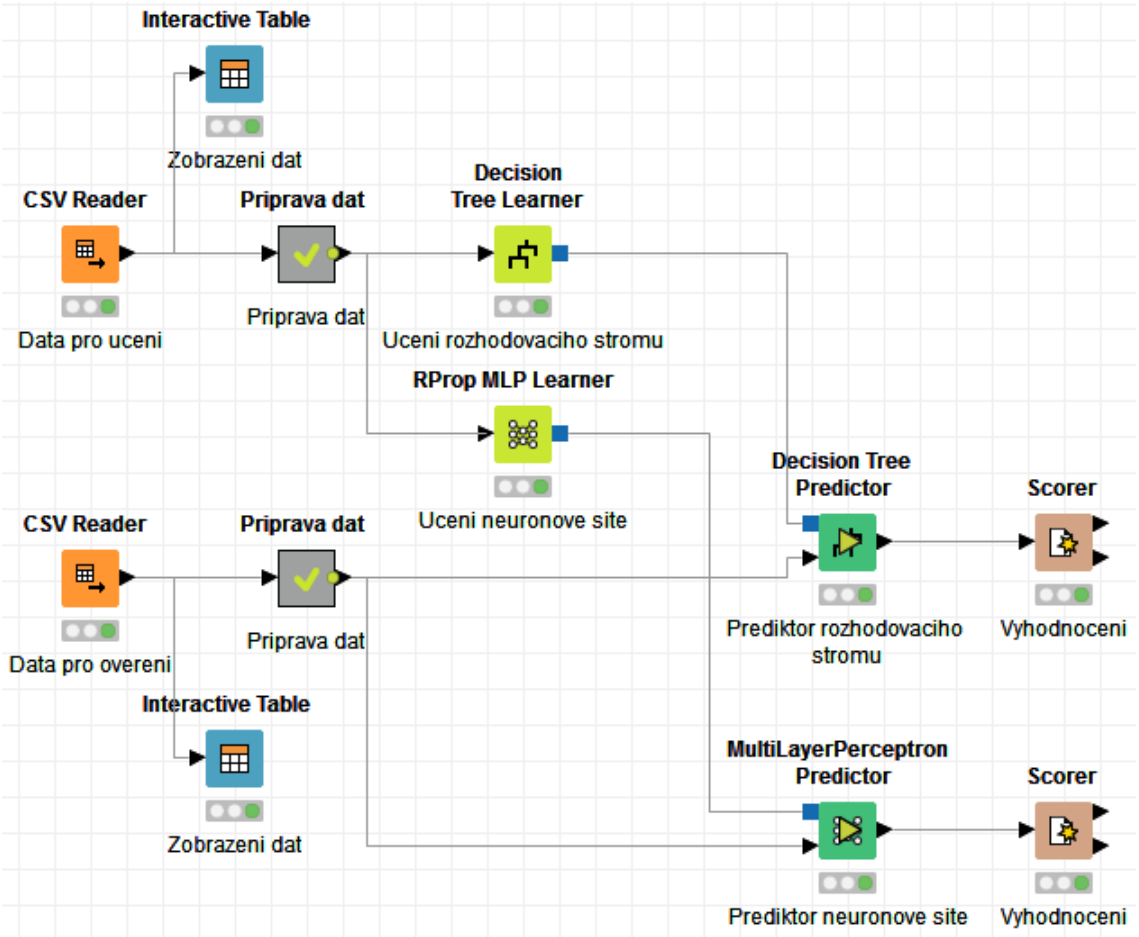


Obrázek 4.7: Příprava dat monitorování provozu strojů v nástroji *IBM SPSS Modeler*

Poznámka: V modelu vytvářeném v prostředí *Knime* bylo pro některé algoritmy, které budou představeny v další části zabývající se tvorbou modelu, potřeba převést cílový atribut na typ *string*, který bude model využívat pro *„učení“*. Důvodem převodu jsou použité algoritmy, které pracují pouze s datovým typem string. Nástroj IBM SPSS Modeler konverzi datového typu nevyžadoval z důvodů automatické konverze.

4.2.4 Model

Model monitorování poruchy strojů se skládá z již zmíněné části přípravy dat a také z rozhodovacích algoritmů. Použité rozhodovací algoritmy jsou rozhodovací strom a neuronová síť. Model je aplikován jak na testovací data, tak i na data pro „*učení*“ pro transformaci vstupů do stejné struktury. Poslední částí modelu je uzel reprezentující *matici záměn* pro vyhodnocení úspěšnosti naučeného modelu. V modelu je dále připojen datový soubor *monitorovaniTestovani*, na kterém je prováděno testování naučeného modelu.



Obrázek 4.8: Model monitorování poruchy strojů v prostředí *Knime*

V následující tabulce je zobrazen přehled naměřených výsledků vytvořeného modelu v základním nastavení, tj. klouzavý průměr atributů *zmena vykonu* a *zmena teploty* byla nastavena na velikost 5. Stejná hodnota byla také nastavena pro atribut *návrat do stabilního stavu uzlu* uzlu *Stav*. Hodnota pro tyto uzly byla zvolena z praktického hlediska, jelikož větší hodnoty pro klouzavé průměry mají za následek vynechání krajních hodnot intervalu a to pro časové řady o několika hodnotách může být velice zavádějící. Model v datovém toku vytvářeném v nástroji *IBM SPSS Modeler* dále obsahuje třemi dalšími algoritmy rozhodovacích stromů. Jelikož *Knime* obdobnými algoritmy nedisponuje, nebyly výsledky zařazeny do následujícího shrnutí a jsou k dispozici na přiloženém CD v souboru *Monitorovani - vysledky modelu*. Tabulka 4.4 obsahuje přehled naměřených hodnot přesnosti testovaných modelů v použitých nástrojích při nastavení uzlů v automatickém režimutj. automatické nastavení stromu a počtu neuronů v neuronové síti.

Tabulka 4.4: Výsledky přesnosti modelu monitorování zkušebního provozu

Rozhod. algoritmus	***IBM SPSS Modeler***	***Knime***
Strom	99,68	99,85
Neuronová síť	99,35	98,23

V následující tabulce je zobrazeno měření na sledování přesnosti modelu pomocí změny počtu vrstev a počtu neuronů neuronové sítě. 4.5

Tabulka 4.5: Vliv počtu neuronů na přesnost modelu

		Úspěšnost [%]	
Počet vrstev	**Počet neuronů**	**IBM SPSS Modeler**	**Knime**
1	1	69,01	96,75
1	3	99,52	65,23
1	5	99,32	96,44
1	10	99,76	98,23
2	1	69,2	56,45
2	3	97,96	38,51
2	5	99,76	69,25
2	10	99,83	93,11

Rozhodovací stromy obsahují několik typů nastavení. Následující tabulky 4.6 a 4.7 měření zobrazují vliv nastavení rozhodovacích stromů na přesnost modelu. Znaménko „-“ označuje výchozí nastavení.

Tabulka 4.6: Nastavení rozhodovacího stromu v prostředí *KNIME*

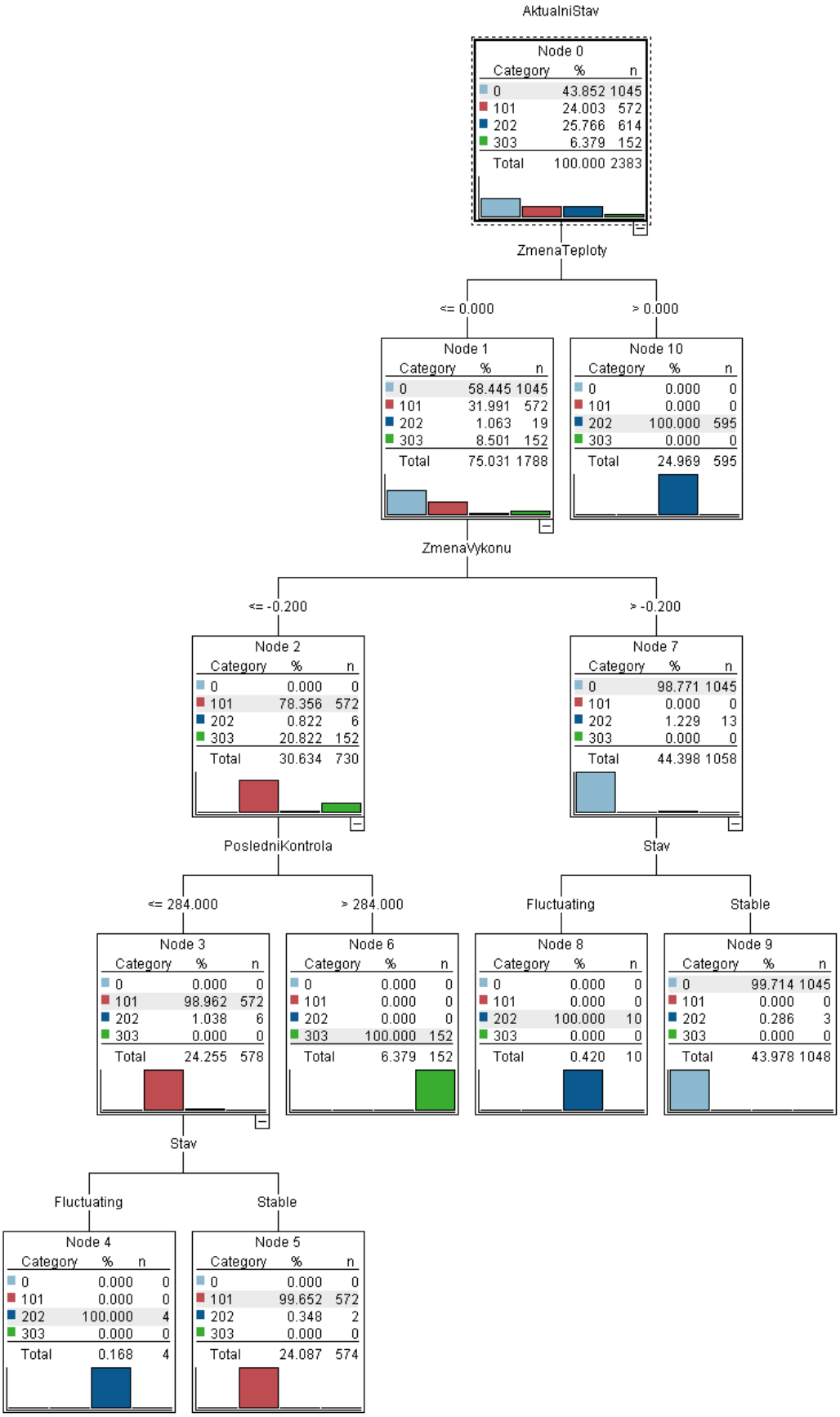
Pruning method	***Přesnost modelu [%]***
MDL	99,78
-	99,85

Tabulka 4.7: Nastavení rozhodovacího stromu v prostředí *IBM SPSS Modeler*

Boosting	**Pruning severity**	**Přesnost modelu** [%]
3	-	99,68
5	-	99,88
10	-	99,88
15	-	99,71
20	-	99,66
-	0	97,11
-	50	97,11
-	75	99,68
-	100	99,68

4.2.5 Závěr

Studie monitorování zkušebního provozu je zaměřena na analýzu a sestavení modelu pro predikci poruchy strojů. Studie obsahuje dva datové soubory obsahující časové řady průběhu činnosti stroje, resp. monitorování vlastností stroje jako jsou tlak, teplota, výkon a další. Datové soubory mají stejnou strukturu, první datový soubor slouží k analýze a natrénování vytvořeného modelu a druhý soubor slouží k následnému testování modelu. Před stavbou predikčního modelu byla analyzována vstupní data pro nalezení závislostí vstupních atributů. Model studie obsahuje rozhodovací strom a neuronovou síť. Rozhodovací strom nabízí přehledový graf, který interpretuje průběh rozhodování a zobrazuje atributy, podle kterých se nejvíce rozhodoval. Rozhodovacího stromu je možné vidět na obrázku 4.9. Nejdůležitějším parametrem pro rozhodování byl atribut změny teploty, který udává, že pokud nastává nárůst teploty, stroj se dostane do chybového stavu 202. Dalším významným atributem pro rozhodování byla změna výkonu stroje. Rozhodovací strom je možné jednoduchým způsobem interpretovat a to segmentací objektů do daných tříd. U neuronové sítě však takováto interpretace není možná. Výsledná přesnost modelu je také ovlivněna nastavením klouzavého průměru spolu s nastavením rozhodovacích uzlů strom a neuronová síť, kde nevhodně zvolené nastavení může mít špatný vliv na výslednou přesnost modelu.



Obrázek 4.9: Rozhodovací strom v IBM SPSS Modeler

4.3 Předpověď spotřeby elektrické energie

Cílem úlohy je sestavit model, který z naměřených dat určí spotřebu na následujících 24 hodin. Data obsahují jak hodnoty spotřeby elektrické energie, tak i teploty z měřených oblastí, kam elektrická energie je distribuována elektrárenskou společností.

4.3.1 Struktura datového souboru

Úloha obsahuje datový soubor ***spotrebaElektriny***. Datový soubor obsahuje 35 064 záznamů spotřeby elektřiny v konkrétních dnech, resp. hodinách spolu s naměřenými teplotami z různých oblastí. V datovém souboru je označeno, zda-li byl v měřený den svátek či nikoliv. Struktura části datového souboru je zobrazena na obrázku 4.10.

	Cas	Spotreba	Celodenni prumerna teplota	Prumerna teplota	Teplota oblast 1	Teplota oblast 2
1	2004-01-01 01:00:00	4921.768	-5.818	-4.949	-5.782	-3.600
2	2004-01-01 02:00:00	4827.295	-5.818	-5.065	-5.906	-3.732
3	2004-01-01 03:00:00	4537.326	-5.818	-5.182	-6.021	-3.916
4	2004-01-01 04:00:00	4478.257	-5.818	-5.321	-6.157	-4.095
5	2004-01-01 05:00:00	4442.755	-5.818	-5.411	-6.280	-4.261
6	2004-01-01 06:00:00	4118.950	-5.818	-5.500	-6.389	-4.368
7	2004-01-01 07:00:00	4256.507	-5.818	-5.561	-6.450	-4.372
8	2004-01-01 08:00:00	3991.896	-5.818	-5.487	-6.399	-4.284
9	2004-01-01 09:00:00	4244.942	-5.818	-5.288	-6.277	-4.054
10	2004-01-01 10:00:00	4478.448	-5.818	-5.123	-6.222	-3.903
11	2004-01-01 11:00:00	4734.789	-5.818	-5.065	-6.275	-3.917

Obrázek 4.10: Rozhodovací strom v IBM SPSS Modeler

Tabulka 4.8 obsahuje popis všech atributů datového souboru studie předpovědi elektrické energie.

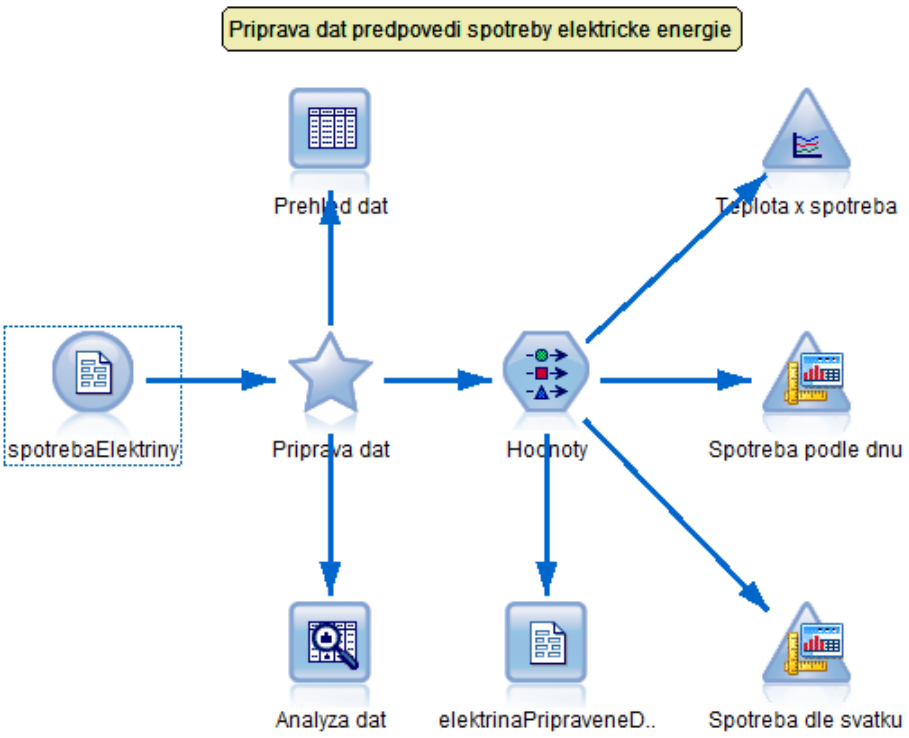
Tabulka 4.8: Atributy datového souboru - Předpověď spotřeby elektrické energie

Název atributu	***Popis***
Cas	Datum a čas provedeného měření
Spotreba	Naměřená spotřeba elektřiny v MW
Celodenni prumerna teplota	Denní průměrná teplota
Prumerna teplota	Průměrná aktuální teplota všech oblastí
Teplota oblast 1	Aktuální teplota oblasti 1
Teplota oblast 2	Aktuální teplota oblasti 2
Teplota oblast 3	Aktuální teplota oblasti 3
Teplota oblast 4	Aktuální teplota oblasti 4
Teplota oblast 5	Aktuální teplota oblasti 5
Teplota oblast 6	Aktuální teplota oblasti 6
Teplota oblast 7	Aktuální teplota oblasti 7
Teplota oblast 8	Aktuální teplota oblasti 8
Svatek	Svátek - hodnoty ano/ne

4.3.2 Příprava dat

Z analýzy atributů vstupního datového souboru, kterou jsem provedl před tvorbou datového modelu, jsem došel k závěru, že v přípravě dat se zaměřím na zjištění informací z poskytovaného data, jako jsou den v týdnu, měsíc apod., díky čemuž bude možné zobrazovat spotřebu elektrické energie v těchto časových údajích. Dále se zaměřím na zjištění výkyvů spotřeby, resp. na zjištění maximální a minimální hodnoty ve stanovených dnech i týdnech spolu se získáním informací o průměrné spotřebě za stanovené období a také zjištění závislosti teploty na spotřebě elektrické energie. Mezi zajímavost přiložím zobrazení závislosti spotřeby elektrické energie na svátcích. Z pořízených dat vyplývá, že data byla naměřena za časový úsek čtyř let.

Pro přípravu dat byl vytvořen vlastní datový tok, ve kterém jsou připravovány data pro tvorbu modelu předpovědi spotřeby elektrické energie. Datový tok je označen jménem ***elektrinaPripravaDat***. Sktruktura datového toku je zobrazena na obrázku 4.11.



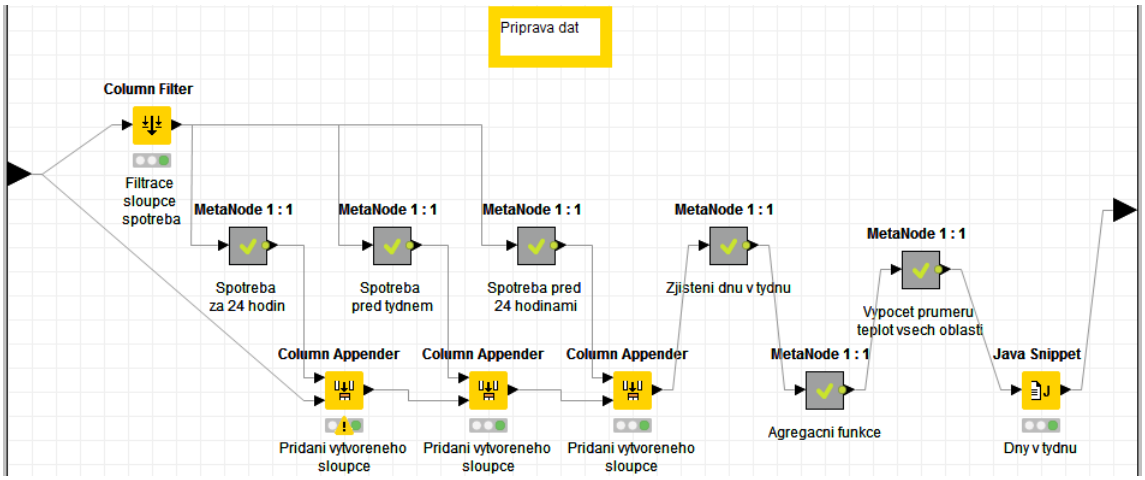
Obrázek 4.11: Datový tok *elektrinaPripravaDat* v nástroji *IBM SPSS Modeler*

Datový tok přípravy dat se skládá z několika uzlů, mezi nimiž je uzel pro načtení datového souboru *spotrebaElektriny*. V datovém toku se nachází uzel pro zobrazení vstupních dat a také uzel ***Data audit***, který slouží k provedení rychlé analýzy bez nutné manipulace s daty. Výstupem uzlu je přehled atributů vstupní matice spolu se zobrazením maximálních, resp. minimálních hodnot číselných atributů, průměrná

hodnota a také celkový počet záznamů, počet zastoupení kategorií atributu apod. Nevýhodou je fakt, že tento typ uzlu se nachází pouze v nástroji *IBM SPSS Modeler* a druhý testovaný nástroj bohužel tímto typem uzlu nedisponuje.

Důležitým uzlem celého datového toku přípravy dat je uzel ***Priprava dat***. Uzel slouží k úpravě vstupních dat do požadovaného tvaru a jedná se o uzel typu ***SuperNode*** v *IBM SPSS Modeleru*, resp. ***Meta node*** v nástroji *Knime*. Uzel slouží k zapouzdření vybraných uzlů do uzlu jednoho. Výhodou nástroje *Knime* je možnost nastavení počtu vstupů a výstupů a tudíž uzel může přijímat více datových toků a následně je zpracovávat pomocí vnitřní logiky navržené uživatelem.

Vnitřní struktura uzlu *Priprava dat* je rozdělena do tří částí. V první části uzlu jsou zjišťovány časové údaje, jako je den v týdnu, měsíc, zjištění spotřeby elektrické energie před 24 hodinami a také zjištění spotřeby minulý týden ke každému pořízenému záznamu. Druhá část uzlů je zaměřena na agregační funkce, tedy na výpočet průměrných hodnot, hledání maximální a minimální hodnoty spotřeby elektrické energie a to jak pro hodnoty denní spotřeby, tak i týdenní spotřebu elektrické energie. V poslední třetí části se počítá průměrná teplota všech oblastí spotřebovávající elektrickou energii. V každé oblasti je počítána průměrná teplota za 24 hodin spolu s průměrnou teplotou za celý týden. V následujícím textu jsou popsány jednotlivé uzly obsažené v ***SuperNode***, resp. ***Meta node*** *Priprava dat*.



Obrázek 4.12: Struktura uzlu *Priprava dat* v nástroji *Knime*

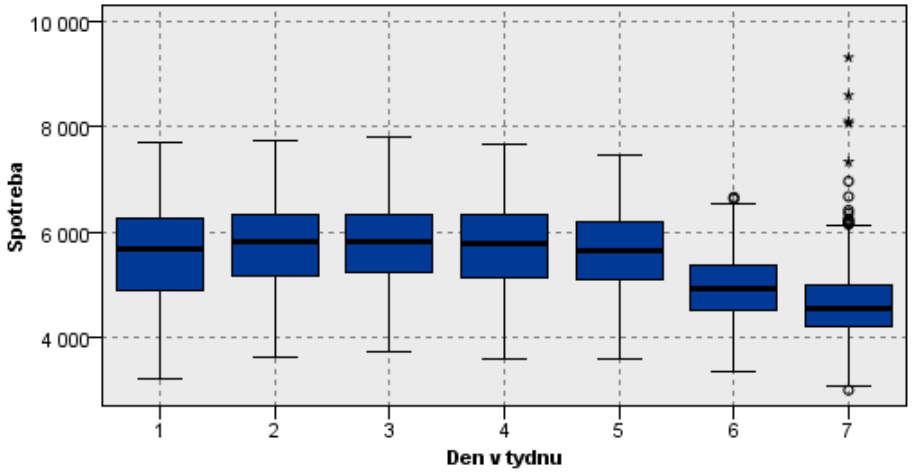
První část uzlu *Priprava dat* je zaměřena na úpravu dat do časových období. To znamená, že jsou v datech vyhledávány údaje o spotřebě před 24 hodinami, spotřeby před týdnem, zjištění dnu a měsíce z časového údaje apod. V nástroji *IBM SPSS Modeler* jsem pro zjištění hodnoty spotřeby před týdnem použil již zmíněný uzel typu *Derive* a pomocí jazyka *CLEM* jsem zapsal příkaz pro posun atributu *Spotreba* o 168 záznamů, abych získal požadovanou hodnotu spotřeby minulého týdne. Pro odvození hodnoty spotřeby elektrické energie, která byla před, resp. za 24 hodin jsem postupoval stejným postupem. Uzel *Derive* s nastavením uzlu *Derive as: Nominal*

jsem dále použil pro zjištění dnu v týdnu z atributu poskytujícího časovou hodnotu. V jazyce *CLEM* jsem použil příkaz *datetime_weekday(Cas)* pro každý možný případ. Výsledkem byl kód o 7 příkazech *datetime_weekday(Cas)*, který byl obdobou konstrukce příkazu *Switch*. Pro zjištění měsíce z časové hodnoty byl použit podobný postup s rozdílem použitého příkazu pro zjištění měsíce.
V nástroji *Knime* jsem musel postupovat jiným způsobem, jelikož takovými funkcemi nedisponuje. Pro spotřeby před 24 hodinami a minulého týdne bylo zapotřebí použít několik typů uzlů, abych získal stejné výsledky jako v druhém nástroji. Mezi operace potřebné pro získání stejných výsledku patří posun hodnot atributu *Spotreba* o požadovaný počet řádků, tj. 24, resp. 168 řádků, vytvoření nového atributu pro zápis posunutých hodnot a následně napárování na původní vstupní matici. To samé platilo pro zjištění dnu v týdnu a měsíce z časového atributu, kdy jsem musel převést časový údaj z typu *string* na typ *datum* pomocí uzlu ***Date Field Extractor*** a následně provést převod hodnot dnu v týdnu za pomocí konstrukce typu *Switch* v uzlu typu *Java snippet* použitím programovacího jazyka *Java* na český formát, jelikož uzel automaticky převádí den v týdnu do tvaru, kdy týden začíná nedělí narozdíl od našeho formátu, kdy týden začíná pondělím. Konkrétní popis implementace je zachycen v podrobném popisu úloh na přiloženém CD.

Druhá část uzlu *Priprava dat* je zaměřena na výpočet průměrných hodnot, hledání maximální a minimální hodnoty spotřeby elektrické energie a to jak pro hodnoty denní spotřeby, tak i týdenních hodnot. Pro tyto účely jsem zvolil v nástroji *IBM SPSS Modeler* uzel typu Derive, do kterého jsem pomocí jazyka CLEM psal požadovanou funkčnost. Pro průměr to byla funkce *@AVE*, pro minimální hodnotu funkce *@MIN*, resp. *@MAX* pro maximální hodnotu. Jelikož *Knime* nedisponoval potřebnými nástroji pro zjištění těchto údajů, musel jsem toto chování uzlů naprogramovat. Z toho důvodu jsem použil již zmíněný uzel *Java snippet*, který umožňuje psát programový kód v jazyce *Java*.

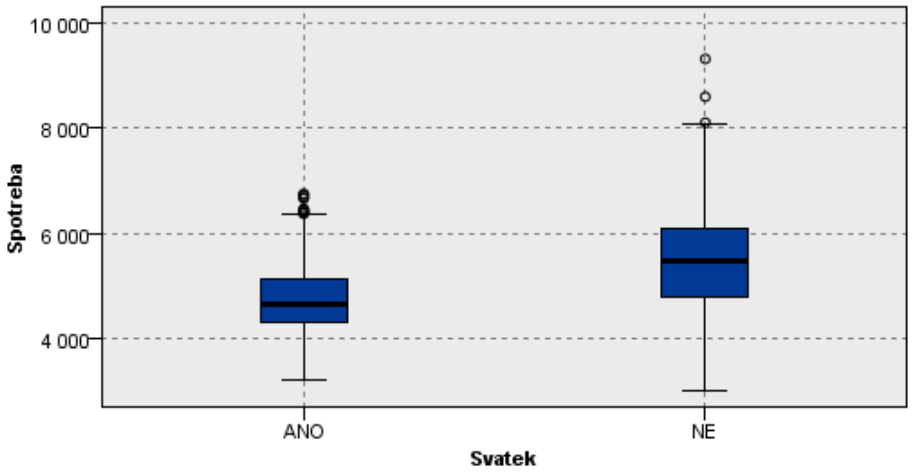
V poslední části uzlu *Priprava dat* počítám průměrnou teplotu všech oblastí, kde se spotřebovává elektrická energie. V každé oblasti je počítána průměrná teplota za 24 hodin spolu s průměrnou teplotou za celý týden.

Dále se v datovém toku nachází tři uzly pro grafické znázornění upravených dat. Mezi tyto uzly patří uzel ***Spotreba podle dnu***, který interpretuje spotřebu elektrické energie ve dnech v týdnu, kterou zobrazuje obrázek 4.13. Výstupem uzlu je krabicový graf zachycující minimální, resp. maximální hodnotu, 1. a 3. kvartil spolu s mediánem a také *outliners*.



Obrázek 4.13: Závislost spotřeby elektrické energie na dnu v týdnu a svátku v nástroji *IBM SPSS Modeler*

Spotřeba elektrické energie je ve všedních dnech, resp. úterý až čtvrtek velice vyrovnaná. V pondělí a pátek je vidět vliv víkendových omezení, kdy může docházet např. k omezení výroby, resp. k její obnově.

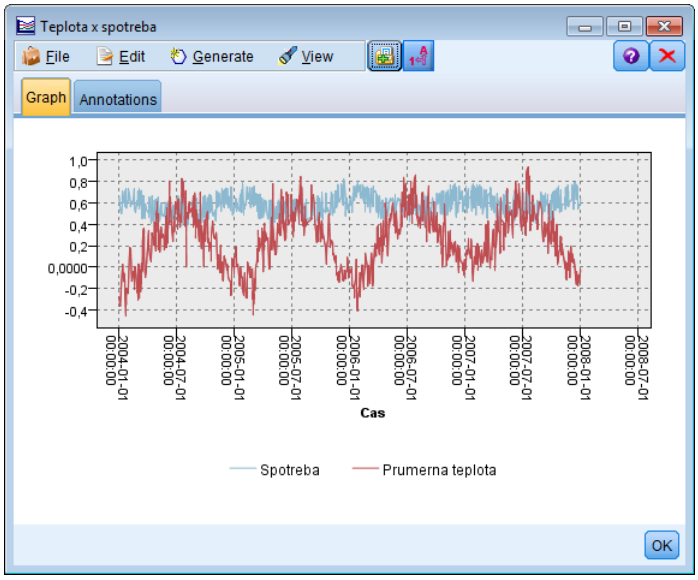


Obrázek 4.14: Závislost spotřeby elektrické energie na dnu v týdnu a svátku v nástroji *IBM SPSS Modeler*

Obrázek 4.14 zachycuje situaci spotřeby elektrické energie při svátcích, která se zjišťuje v uzlu ***Spotreba dle svatku***. Při svátcích má mnoho obchodů, státních firem a dalších institucích zavřeno a tudíž nedochází k tak „*velkému*“ odběru jako v pracovní den.

Poslední uzel datového toku zabývající se přípravou dat je uzel ***Teplota x spotreba*** a slouží k zachycení závislosti spotřeby elektrické energie na teplotě, kterou

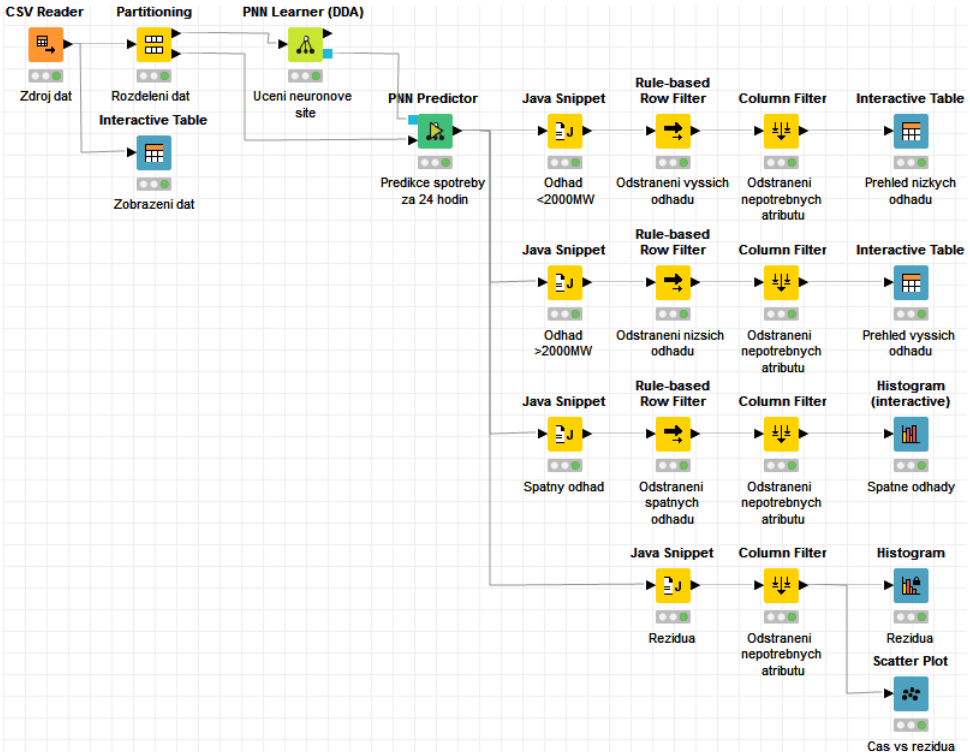
zachycuje obrázek 4.15 na průměrné teplotě. Z obrázku je patrné, že při poklesu venkovní teploty stoupá spotřeba elektrické energie. V případové studii jsou použity záznamy spotřeby elektrické energie pořízené v roce 2004. V současné době odhaduji, že nastává opačná situace, kdy při rostoucí teplotě roste spotřeba elektrické energie. Důvodem je rostoucí trend chladících zařízení, jako jsou klimatizace apod. a tím svázaný růst spotřeby energie.



Obrázek 4.15: Závislost spotřeby elektrické energie na průměrné teplotě v nástroji *IBM SPSS Modeler*

4.3.3 Model předpovědi spotřeby elektrické energie

Další část úlohy je zaměřena na vlastní modelování předpovědi spotřeby elektrické energie za dalších 24 hodin. Pro tyto účely jsem vytvořil datový tok s názvem ***elektrinaModel***, ve kterém je realizována předpověď. Následující obrázek 4.16 zobrazuje strukturu datového toku modelu budoucí spotřeby elektrické energie.



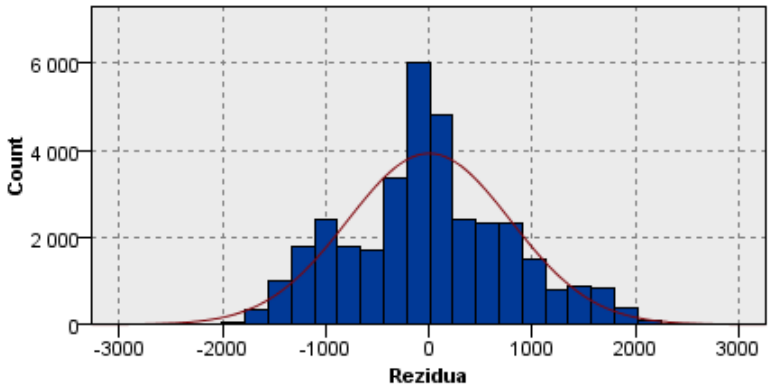
Obrázek 4.16: Model předpovědi spotřeby elektrické energie v nástroji *Knime*

Prvním uzlem ve datového toku je uzel pro načtení dat, které jsem uložil v přípravné části této úlohy. Dále je potřebovat rozdělit načtená data na trénovací a testovací množinu tak, aby měl model data pro učení a testování. K tomuto rozdělení jsem použil uzel ***Rozdeleni dat*** typu ***Partition***, ve kterém jsem nastavil relativní rozdělení dat na 50%. Takto jsem docílil rozdělení připravených dat na dvě stejně velké části. Pro predikci pomocí neuronové sítě jsem vybral uzel neuronové sítě a jako cílovou proměnnou jsem nastavil atribut *Spotreba za 24 hodin*.

V poslední části modelu jsem filtroval špatné výsledky predikce a to především, když v modelu dojde k tzv. *outliers*, tedy k hodnotám, které se rapidně liší od skutečných hodnot. Za tímto účelem byly vytvořeny tři větve, kde v této první větvi zjišťuji odhady, které se lišili o více než 2000MW a zároveň byl odhad menší než skutečná hodnota. Druhá větev obsahuje přehled předpovědí, které byly větší než skutečná hodnota a měly rozdíl větší než 2000MW.

Dále jsem označil špatné předpovědi, které se lišily o max. 500MW pomocí uzlu *Spatny odhad*.

Pro zobrazení reziduí predikované hodnoty se skutečnou hodnotou v dalších 24 hodinách slouží histogram s názvem Rezidua, který zobrazuje četnosti jednotlivých rozdílů, viz. obrázek 4.17.



Obrázek 4.17: Histogram reziduí v nástroji *IBM SPSS Modeler*

4.3.4 Výsledky

Z výsledků je možné usoudit, že k nejvíce špatným předpovědí docházelo při vyšší změně teploty v jinak konstantních měsících. V nástroji *IBM SPSS Modeler* se jednalo se pouze o měsíc Říjen, kdežto výsledky v nástroji *Knime* ukazovali zastoupení více měsíců. Celkově však šlo o období náchylné na změny spotřeby z důvodu nastupujícího zimního období. Model celkově produkoval 10 nízkých odhadů v nástroji *IBM SPSS Modeler*, resp. 74 v *Knime*. Jedná se tzv. odhady, které byly nižší než skutečná hodnota za 24 hodin, a jejich rozdíl byl větší než 2000MW.

Pro vysoké odhady, tj. odhady vyšší o 2000MW než byla skutečná spotřeba elektrické energie za 24 hodin, model produkoval 3 předpovědi v nástroji *IBM SPSS Modeler* a 75 předpovědí v nástroji *Knime*. Pro tuto úlohu byla zvolena neuronová síť typu *PNN* a to z důvodu, že neuronová síť použitá v předešlé úloze nepodporuje jiné vstupní parametry, než je typ *double* (kromě cílové proměnné). Výsledkem jsou naměřené hodnoty, které se neshodují s výsledky v nástroji *IBM SPSS Modeler*. Nejenom že výpočet byl časově náročný (desítky minut), ale také přesnost modelu nedosahovala podobných výsledků jako v nástroji *IBM SPSS Modeler*. Nepřesnost této neuronové sítě byla také ověřena v předešlé případové studii, kdy síť dosahovala přesnosti pouze 42%. Pro vyvození závěru o použití této neuronové sítě v nástroji *Knime*, by bylo zapotřebí provést testování na dalších sérií úloh. Ze zkušeností v práci v *IBM SPSS Modeler* vím, že neuronová síť je silný a spolehlivý nástroj v tomto nástroji. Platforma *Knime* je typu *open source* a je možné, že jeho některé moduly jsou stále vývoji a mohou obsahovat chyby.

Histogramem jsem zjistil, že k nejvíce špatným předpovědím docházelo v neděli. Důvodem je zřejmě fakt, že v neděli docházelo k největším rozptylům spotřeby elektrické energie.

4.3.5 Závěr

Studie předpovědi spotřeby elektrické energie je zaměřena na analýzu a předpověď na základě záznamů časové řady, které byly pořízeny v časovém horizontu čtyř

let. Pomocí data miningových nástrojů byly z poskytnutých dat zjištěny závislosti spotřeby elektrické energie na svátcích, dnech v týdnu a také byla zjištěna závislost spotřeby na venkovní teplotě. Pomocí analýzy byl následně vytvořen model pro ověření přesnosti předpovědi na testovacích datech. Modelování předpovědi spotřeby elektrické energie je nicméně velmi složitá úloha a nelze dosáhnout přesných výsledků, ale pouze přibližných hodnot v určitém rozmezí. Hlavním důvodem je vysoká variabilita teploty, což velmi ovlivňuje vlastní predikci. Výsledky je nutné brát s rezervou.

4.4 Hodnocení použitých nástrojů

Pro tvorbu modelů případových studií byly použity datamingové nástroje *IBM SPSS Modeler* a *Knime*. *IBM SPSS Modeler* byl testován ve zkušební verzi 17 a nástroj *Knime* byl použit ve verzi 3.0.1. v plném balíčku, tzv. obsahoval všechny volně dostupné doplňkové rozšíření. Oba použité nástroje byly testovány v 64 bitové verzi na operačním systému Windows 10.

V současné době je k dispozici *IBM SPSS Modeler* ve verzi 18, která nabízí několik novinek a to především implementací algoritmů, které byly dostupné pouze na serverové straně a nebyly implementovány v klientu jako je například *Linear Support Vector Machines*, *Linear-AS* pro linerání regresi, *Tree-AS*, který je založen na algoritmu CHAID, a další. Velkým přínosem je také možnost paralelních běhů těchto algoritmů a použít tak více jader procesoru pro výpočet. Nově je možné používat programovací jazyk *Python* s rozšířením *Spark*, které slouží pro programování s využitím výpočetních clusterů. Dříve byla tato možnost dostupná pouze na výpočetním serveru *AS* - analytic server, ke kterému se muselo pomocí *IBM SPSS Modeler* připojit. Připojení však neumožňovala základní verze software, ale pouze nástroje s „*vyšší*“ licencí. Velkou změnou prošel nástroj v propojení s komunitou. Nástroj se v poslední verzi snaží být více otevřen komunitě a nyní umožňuje pomocí kontextové nabídky uživateli zobrazit dostupná rozšíření, které lze dodatečně implementovat. [18]

Data miningový nástroj *Knime* je oproti svému konkurentovi aktualizován častěji a to především aplikací oprav vyskytujících se chyb. Poslední aktualizací nástroje je *Knime* ve verzi 3.3, která vyšla 6.12.2016. Nová verze obsahuje změny především v rychlosti zpracování dat a to především při práci s datově obsáhlými daty uloženými v Excelu. Vylepšení se také týkalo rozšířeních poskytujících *Text Mining*. V současné verzi uzly poskytující *Text Mining* umožňují zpracovávat formáty typu *pdf*, *ppt*, *doc*, *txt*, *zip* a další, a navíc disponují extrakcí metadat zkoumaných souborů, jako je autor, čas modifikace dokumentu apod. Další novinkou v oblasti *Text Mining* je rozpoznání jazyka psaného textu. Příjemná nová funkce je v dnešní době především v implementaci uzlů pro vzdálené připojení do cloudových uložišť, mezi které patří ***Amazon S3*** a ***Azure Blob Store***, a následné zpracovávání souborů z těchto webových služeb. Mezi další změny patří aktualizace platformy Eclipse, na které je *Knime* založen. Platforma byla aktualizována na verzi Neon (4.6). Poslední z velkých

změn je automatická instalace chybějících rozšíření při práci s *workflow* z novějších verzí nástroje, nebo při použití práci s *workflow*, které obsahují některý uzel z komunitního uložiště. [19]

Nyní se zaměřím na hodnocení použitých nástrojů z praktického hlediska, které vyšlo z vytváření představených studií. První dojem každého programu utváří prostředí aplikace. Prostředí obou nástrojů má velmi podobné členění a skládá se z hlavního okna pro tvorbu datového toku a menu obsahující uzly, které se pro datový tok používají. Uzly jsou seskupeny do kategorií, ve kterých nástroj *Knime* nabízí vyhledávání podle zadaného textu. Vyhledávání nástroj *IBM SPSS Modeler* nenabízí, což pro početné skupiny uzlů je značnou nevýhodou.

Hodnocení je také zaměřeno na hodnocení funkcí, které nástroje nabízejí. *IBM SPSS Modeler* nabízí bohatou řadu vestavěných funkcí a to včetně komerčních algoritmů jako je např. klasifikační algoritmus *C5.0*. Nástroj *Knime* ve směru „*vybavenosti*“ funkcemi zaostává a nabízí pouze funkce základní, nebo řadu modifikovaných algoritmů. Nevýhodou některých algoritmů je i doba výpočtu, která může být oproti komerčnímu nástroji několikanásobně vyšší, což se například projevilo ve studii zabývající se spotřebou elektrické energie, kdy výpočet neuronové sítě trval až desítky minut, kdežto v komerčním nástroji *IBM SPSS Modeler* byl výpočet dokončen během necelé minuty. Použití neuronové sítě se v nástroji *Knime* příliš neosvědčilo, proto doporučuji použití jiného algoritmu podle typu úlohy. Rozdíl v obsažených algoritmech je především licenčnímu omezení. V tomto směru je nutné zopakovat, že nástroj *Knime* je v základní verzi distribuován zdarma, kdežto *IBM SPSS Modeler* je nabízen pouze v komerčních licencích viz. 3.2. Naopak velkou výhodou nástroje *Knime* je možnost vytváření doplňujících rozšíření. Rozšíření mohou obsahovat nové algoritmy, uzly, nebo celé předpřipravené datové toky. V tomto směru je velice aktivní komunita, která již vytvořila spoustu rozšíření.

Srovnání použitých nástrojů z hlediska složitosti datového modelu je patrná z vytvořených případových studií popisovaných v této kapitole. Stejná úloha v nástroji *IBM SPSS Modeler* jde jednodušeji vytvořit pomoci několika uzlů než je tomu v nástroji *Knime*. Práce v obou nástrojích je intiutivní, ovládání je velice vlídné. Výsledné řešení práce s daty, modelování i evaluace či nasazení je složeno z dostupných uzlů (bloků) spojených orientovanými spojnicemi, které naznačují tok daty mezi uzly. V nástroji *IBM SPSS Modeler* je to *stream* a *workflow* v nástroji *Knime*. V obou nástrojích je nutné porozumět jednotlivým uzlům (blokům) tak, aby jejich použití bylo smysluplné. Některé uzly umožňují velkou variabilitu nastavení a výběr vstupních parametrů od jednoduchých či expertních, které vyžadují od dataminera hlubší znalosti data miningové algoritmizace. V různých blocích je možné *„doprogramovat“* vlastní implementaci. Složitější část nastává při tvorbě programových uzlů pokud prostředí nenabízí požadovanou funkčnost a to ve formě programového kódu. *Knime* umožňuje v k tomu určených uzlech používat programovací jazyk *JAVA* nebo jazyk *R*. Tuto možnost jsem hojně využíval při tvorbě případových studií. *IBM SPSS Modeler* možností programování v jazyce *R* také disponuje, nicméně pro

používání některých uzlů je nutné znak skriptovací jazyk *CLEM*, který se používá například v již zmíněném uzlu *Derive* použitém v případových studií. Oproti jazyku *JAVA* se jedná o psaní výrazů, podmínek a podobných konstrukcí. Jazyk *CLEM* je proprietární jazyk nástroje *IBM SPSS Modeleru* a tudíž není běžnou znalostí IT specialistů.

Důležitým srovnáním obou nástrojů je nutné věnovat grafické interpretaci výstupů. V tomto směru jasně převládá *IBM SPSS Modeler*, se kterým se nástroj *Knime* nemůže srovnávat. Rozdíly v reprezentaci výstupů lze vidět v přiložených obrázcích této kapitole, nebo v příručkách vytvořených studií v přiložené příloze. *IBM SPSS Modeler* nabízí výstup v přehledných grafech, které nabízejí řadu nastaveních a to od nastavení barev přes omezení vykreslované oblasti až po interakci s grafem. *Knime* podobnými nastavení disponuje také, nicméně grafický výstup a jednoduchost ovládání neodpovídá zvyklostem dnešní doby, což považuji v současné době za velkou nevýhodu tohoto nástroje. Vylepšení grafické části pak mohou poskytovat některé nabízené rozšíření nebo export a následné zpracování jinými programy jako je Matlab. Výstup v grafické podobě je také umožněn programovat pomocí jazyka *R*.

Závěrem bych doporučil nástroj *Knime* pro uživatele, kteří již mají zkušenosti s ostatními data miningovými nástroji a nebojí se experimentovat a vytvářet vlastní uzly s požadovaným chováním pomocí vlastních implementací a to především pomocí programovacího jazyka ať už se jedná o jazyk *JAVA*, *R*, nebo práci s externími programy jako je Matlab. Na druhou stranu nástroj *IBM SPSS Modeler* bych doporučil uživatelům, kteří vyžadují kvalitní grafický výstup, upřednostňují jednoduché ovládání a požadují rychlé řešení problému.

5 Závěr

Cílem diplomové práce bylo vytvořit případové studie, které obsahují časové řady a sestavit predikční modely, které budou modelovány v data miningových nástrojích používaných ve výuce spolu se srovnáním použitých nástrojů z hlediska použití a funkcí, které nabízejí.

Před samotnou tvorbou predikčních modelů bylo zapotřebí uvést čtenáře do problematiky časových řad a jejich členění. Shrnuty byly základní pojmy, charakteristiky a manipulace s časovými řadami.

Kapitola 3 je věnována popisu využití časových řad v data maningu spolu se základním uvedením čtenáře do problematiky této analýzy. V textu jsou dále uvedeny popisy vybraných data miningových nástrojů spolu s komerčním nástrojem *IBM SPSS Modeler* a s volně dostupným nástrojem *Knime*, které byly vybrány a detailněji popsány z důvodu použití ve výuce předmětu zabývajícím se problematikou *dolování dat*.

Stěžejní část diplomové práce tvoří případové studie, které byly vytvořeny za účelem ukázky zpracování časových řad ve vybraných data miningových nástrojích. V této části jsou také popsány algoritmy, které se ve vytvořených studií používají. Mezi použité algoritmy patří rozhodovací strom a neuronová síť. První případová studie popisuje monitorování zkušebního provozu. V případové studii monitorování zkušebního provozu jsou zaznamenávány vlastnosti strojů, jako je tlak, teplota výkon a další. Měřené veličiny slouží pro sestavení modelu a následnou predikci poruchy strojů. Druhá případová studie je zaměřena na analýzu a predikci dodávek spotřeby elektrické energie elektrárenské společnosti na následující den. V obou případech byly vytvořeny datové toky spolu s detailními příručkami, které popisují vytváření případových studií a mohou složit jako podklad pro výuku.

V poslední části diplomové práce se zabývám hodnocením použitého softwaru. Pro budování datových modelů byly vybrány dva data miningové nástroje a to *IBM SPSS Modeler* a *Knime*. Důvodem výběru je používání těchto nástrojů ve výuce data miningu a také ukázat čtenářům jejich srovnání při vytváření případových studií. Srovnání má případnému řešiteli data miningového projektu ulehčit rozhodování o DM nástroji, který chce použít. Porovnání data miningových nástrojů bylo uskutečněno z několika pohledů. Nástroje byly podrobeny hodnocením nabízených funkcí, kde *IBM SPSS Modeler* nabízí uzly, které obsahující více vestavěných funkcí a

algoritmů než je tomu u druhého nástroje. Mezi algoritmy patří i komerční algoritmy jako je například rozhodovací strom *C5.0* a jímž nedisponuje nástroj *Knime*. Při zpracování případových studií se mi použití neuronové sítě v nástroji *Knime* příliš neosvědčilo, proto doporučuji použití jiného algoritmu podle typu úlohy. Hlavním rozdílem obou nástrojů je licenční politika. *IBM SPSS Modeler* je komerční nástroj nabízený v několika licencích, které byly popsány v kapitole 3.2, kdežto nástroj *Knime* je v základní verzi volně dostupný a je možné do něj vytvářet vlastní doplňky, nebo je lze získat od aktivní komunity. Nástroje kladně hodnotím po stránce tvoření datových modelů z důvodu intuitivní tvorby datových toků používáním dostupných funkcí. Nevýhodou je fakt, že *Knime* ve srovnání s nástrojem *IBM SPSS Modeler* mnoha funkcemi nedisponuje a ve složitějších případech je nutná vlastní implementace pomocí programovacího jazyka *JAVA* nebo *R*, což v případových studiích bylo v mnoha případech nutné. Z hlediska grafické reprezentace výstupu *IBM SPSS Modeler* vyniká a z mého hlediska není nástroj *Knime* v této kategorii soupeřem. V testované verzi je značné, že *Knime* nemůže konkurovat ani s možným využitím externích nástrojů jako je *Matlab* apod. V závěru hodnocení doporučuji nástroj *Knime* pro uživatele, kteří již mají zkušenosti s ostatními data miningovými nástroji a nebojí se experimentovat a vytvářet vlastní uzly s požadovaným chováním pomocí vlastních implementací a to především pomocí programovacího jazyka ať už se jedná o jazyk *JAVA*, *R*, nebo práci s externími programy jako je Matlab. Na druhou stranu nástroj *IBM SPSS Modeler* doporučuji uživatelům, kteří vyžadují kvalitní grafický výstup, upřednostňují jednoduché ovládání a požadují rychlé řešení problému.

Literatura

[1] KVASNIČKA, Michal a Osvald VAŠÍČEK. *Úvod do analýzy časových řad* [online]. 2001 [cit. 2015-11-10]. Dostupné z: http://www.econ.muni.cz/~qasar/_vyuka/emm2/skriptaemmii.pdf

[2] BERKA, Petr. *Dobývání znalostí z databází*. Vyd. 1. Praha : Academia, 2003. 366 s. ISBN 8020010629.

[3] ARLT, Josef, Markéta ARLTOVÁ a Eva RUBLÍKOVÁ. *Analýza ekonomických časových řad s příklady*. Vyd. 1. Praha: Vysoká škola ekonomická, 2002, 147 s. ISBN 80-245-0307-7

[4] HANČLOVÁ, Jana a Lubor TVRDÝ. *Úvod do analýzy časových řad* [online]. Ostrava, 2003 [cit. 2015-11-03]. Dostupné z http://www.gis.vsb.cz/pan-old/Skoleni_Texty/TextySkoleni/AnalyzaCasRad.pdf

[5] *Orange Data Mining* [cit. 2016-04-12]. Dostupné z: http://www.predictiveanalyticstoday.com/orange-data-mining/

[6] *Weka Data Mining* [cit. 2016-04-12]. Dostupné z: http://www.predictiveanalyticstoday.com/weka-data-mining/

[7] *Rattle: A Graphical User Interface for Data Mining using R* [cit. 2016-04-12]. Dostupné z: http://rattle.togaware.com/

[8] *Rapidminer* [cit. 2016-04-12]. Dostupné z: http://www.predictiveanalyticstoday.com/rapidminer/

[9] *IBM SPSS Modeler* [cit. 2016-04-12]. Dostupné z: http://acrea.cz/software/ibm-spss-modeler/

[10] Product pricing, *IBM SPSS Modeler Marketplace* [cit. 2016-04-12]. Dostupné z: https://www.ibm.com/marketplace/cloud/spss-modeler/purchase/us/en-us#product-header

[11] *FAQ* [cit. 2016-04-12]. Dostupné z: https://tech.knime.org/faq#q1

[12] *SPSS Modeler 17.0 Documentation* [on-line], [cit. 2016-07-08] Dostupné z: http://www-01.ibm.com/support/docview.wss?uid=swg27043831

[13] *Decision Tree Learner* [cit. 2016-09-28]. Dostupné z: https://www.knime.org/files/nodedetails/_mining_dtree_Decision_Tree_Learner.html

[14] Mrázová, Iveta, *Dobývání znalostí* [on-line], 2011 [cit. 2016-12-03]. Dostupné z: http://ksvi.mff.cuni.cz/~mraz/datamining/lecture/Dobyvani_Znalosti_Prednaska_Uvod.pdf

[15] Berka, Petr, *Dobývání znalostí z databází (kap. 5.1)* [on-line], 2009 [cit. 2016-08-19] Dostupné z: http://sorry.vse.cz/~berka/docs/izi456/kap_5.1.pdf

[16] RYCHLÝ, Marek, *Klasifikace a predikce* [on-line], 2005 [cit. 2016-08-20] Dostupné z: http://www.fit.vutbr.cz/~rychly/public/docs/classification-and-prediction/xhtml/classification-and-prediction.xhtml#dectrees

[17] MAYER-SCHÖNBERGER, Viktor a Kenneth CUKIER. *Big Data*. Brno: Computer Press, 2014. ISBN 978-80-251-4119-9.

[18] FISCHER, Ted, *Announcing IBM SPSS Modeler 18* [on-line], 15.3.2016 [cit. 2016-18-12] Dostupné z: https://developer.ibm.com/predictiveanalytics/2016/03/15/announcing-ibm-spss-modeler-18/

[19] *KNIME Analytics Platform 3.3 released* [on-line], 6.12.2016 [cit. 2016-18-12] Dostupné z: https://www.knime.org/about/news/knime-analytics-platform-33-released

[20] *KNIME Product Matrix* [on-line], [cit. 2016-25-12] Dostupné z: https://www.knime.org/products/product-matrix#Extensions

Seznam tabulek

Seznam obrázků