



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Monitorování provozu strojů

Knime

Dataminingová úloha

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

Autor práce: **Bc. Jiří Kratochvíl**



1. Úvod

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 byl zvolen rozhodovací strom a neuronová síť. Celá tato úloha bude tvořena na volně dostupné platformě **Knime**.

2. Data

Monitorování zkušebního provozu obsahuje dva datové soubory. První z nich je datový soubor pro „učení“ modelu a je označen jako **monitorovaniUceni**, nebo-li 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 *Obrázek 1: přehled vstupních dat*.

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 1: přehled vstupních dat

V následující tabulce jsou popsány jednotlivé atributy zkoumaných souborů.

Tabulka 1 přehled atributů vstupního souboru

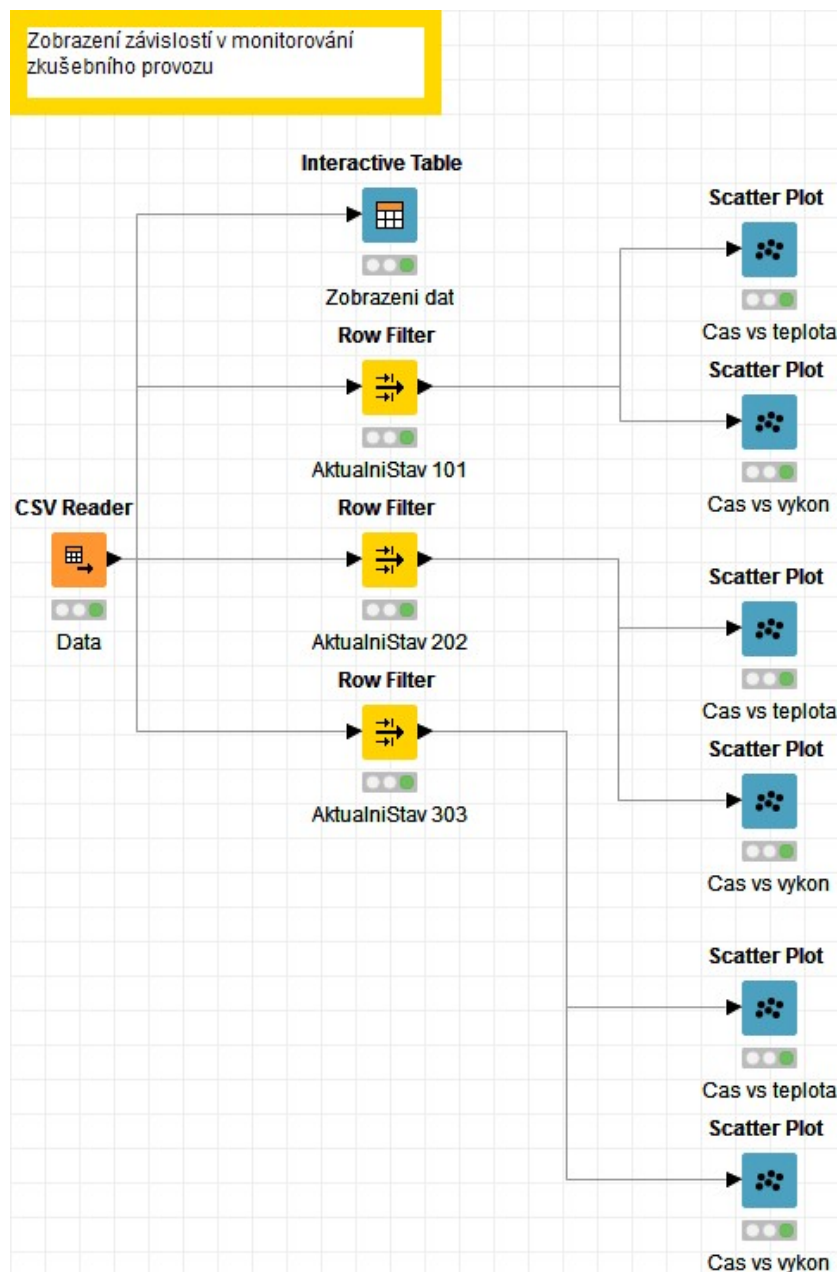
Název	Akce
Cas	čas pořízeného záznamu
Vykon	výkon stroje

Teplota	teplota stroje
Tlak	tlak stroje, hodnota 0 pro normální stav, hodnota 1 pro výstražný stav
PosledniKontrola	čas od provedené poslední kontroly
KoncovyStav	chybový stav, do kterého se dostane stroj na konci časové řady
AktualniStav	stav stroje, 0 normální stav (101, 202 a 303 jsou chybové stavy)

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

3. Analýza vstupních dat

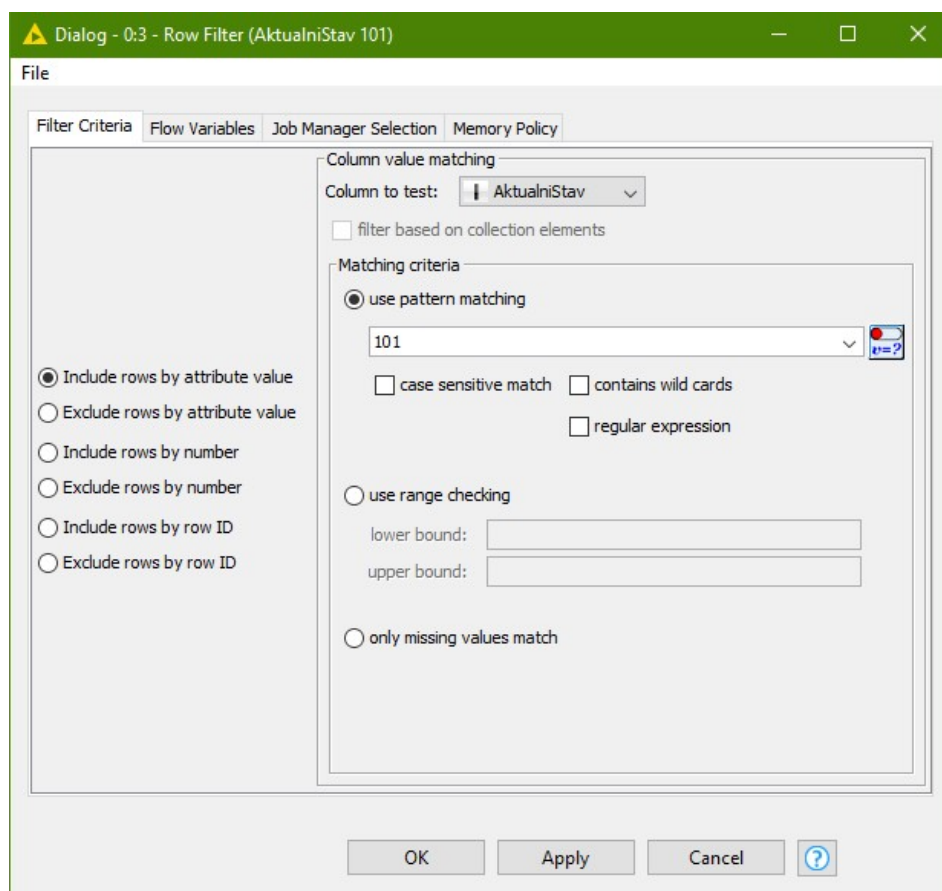
Pro zjištění závislostí mezi atributy byl zvolen datový soubor *monitorovaniUceni*. Za účelem analýzy závislostí vstupních dat jsem vytvořil *workflow* s názvem ***Monitorovani_zavislosti***. Cílem analýzy závislosti atributů je zjistit chování stroje v chybových stavech a odhalit atributy, které nejvíce ovlivňují stroj, resp. popsat charakteristiky chybových stavů k atributům teploty a výkonu. 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 2: zobrazení závislostí sledovaných veličin*.



Obrázek 2: zobrazení závislostí sledovaných veličin

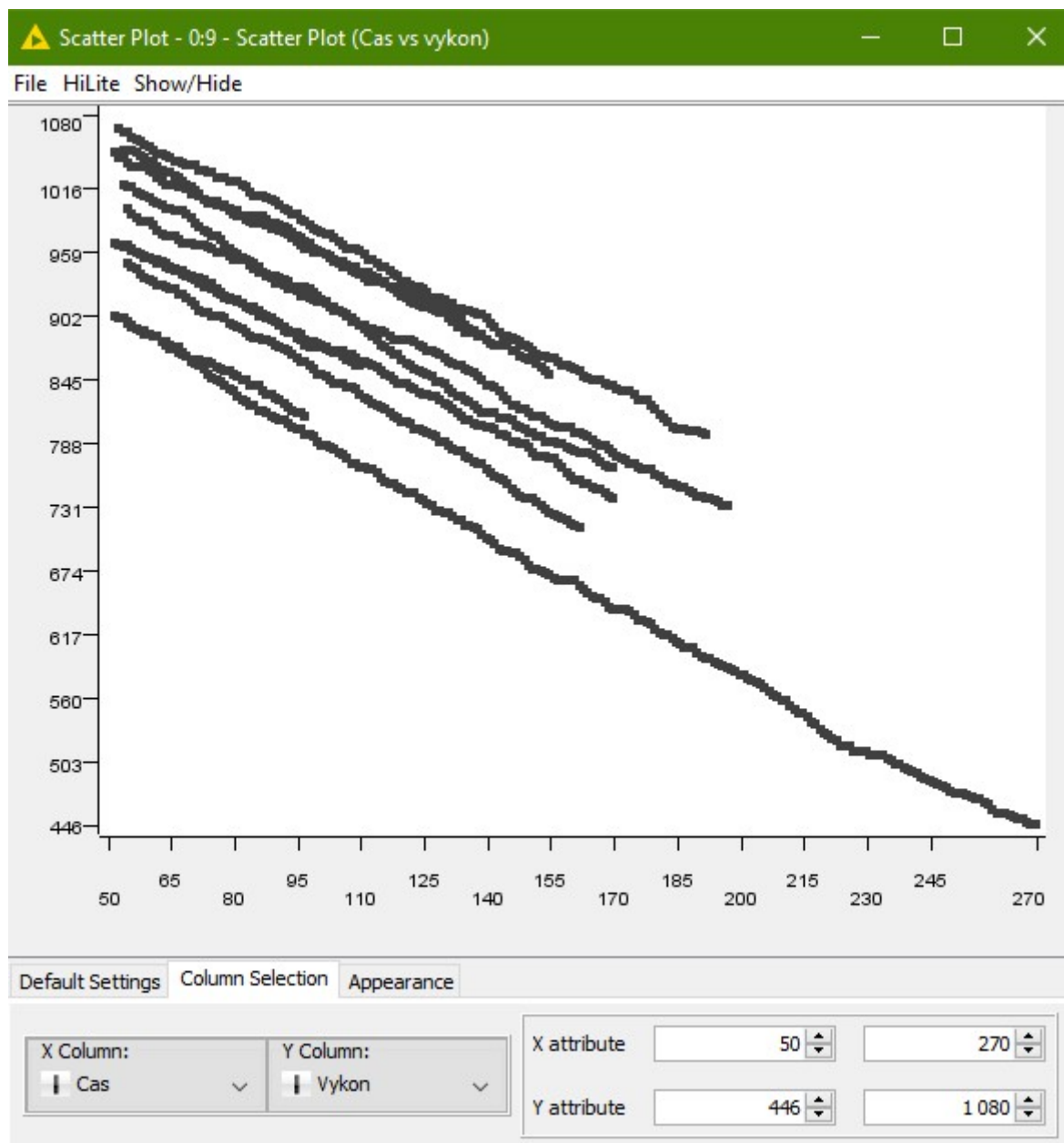
Workflow je složen z několika uzlů. Uzel **CSV Reader** slouží k načtení dat, v našem případě k načtení souboru *monitorovaniTestovani*. Ke správnému načtení tohoto souboru je nutné v nastavení uzlu odškrtnout volbu *Has Row Header*. Důvodem je, že nástroj Knime by vzal první atribut *Time* jako identifikátor řádku, a protože se v datovém souboru objevuje několik časových řad (měření), které vždy začínají od nuly, tak by se objevovalo mnoho duplicitních údajů a uzel by z tohoto důvodu nebyl schopen tato data načíst.

Další uzel, s názvem **Interactive Table**, slouží k zobrazení dat. V našem případě je uzel připojen na již zmiňovaný uzel *CSV Reader* a díky tomu jsme schopni zobrazit námi načtená data. Dalším důležitým uzlem je uzel **Row Filter**. Uzel filtruje data pomocí námi zvoleného kritéria. V tomto případě jsem chtěl zjistit závislost atributu **AktualniStav** na ostatních atributech a z toho důvodu byly vytvořeny tři uzly *Row filter*, protože soubor obsahuje tři chybové stavy, které mohou nastat. Konkrétní nastavení uzlů je zobrazeno na následujícím obrázku. Uzly se liší pouze chybovými kódy, které chceme filtrovat, tedy hodnoty 101, 202 a 303.



Obrázek 3: nastavení uzlu row filter

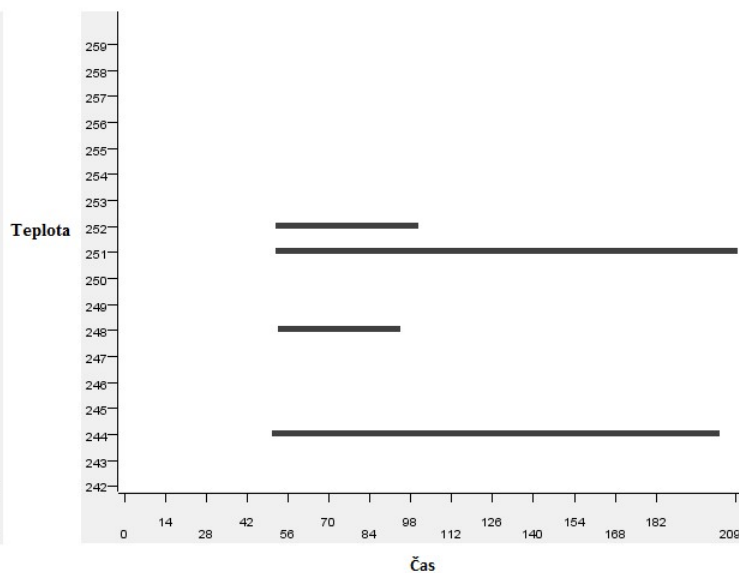
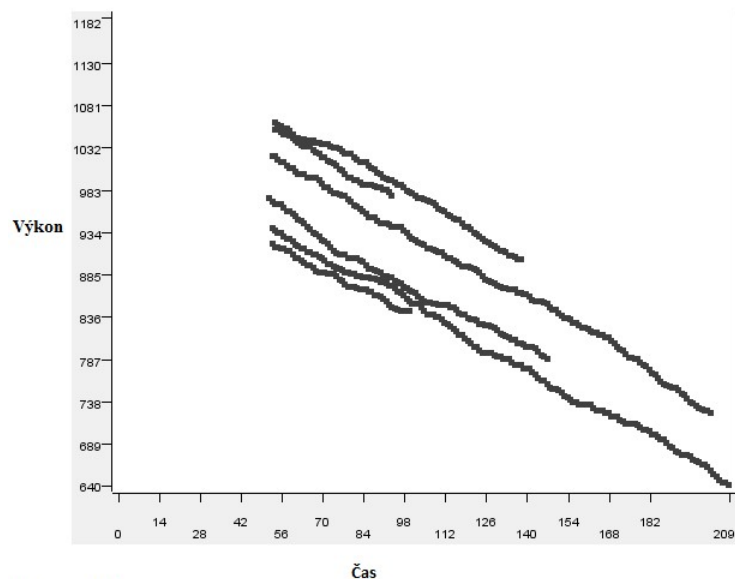
Posledním uzlem *workflow* v části úlohy analýzy dat je uzel **Scatter Plot**, který slouží k vykreslení závislosti dvou vybraných veličin. Vykreslením závislostí v uzlu *Scatter Plot* budeme rozumět 2D graf, kde na ose x a ose y lze vybrat atributy z datového souboru. V tomto uzlu je také možno manipulovat s rozsahem hodnot os, tudíž si můžeme nechat zobrazit pouze určitou část grafu, která nás nejvíce zajímá. Nastavení uzlu *Scatter Plot* je zobrazena na následujícím obrázku.



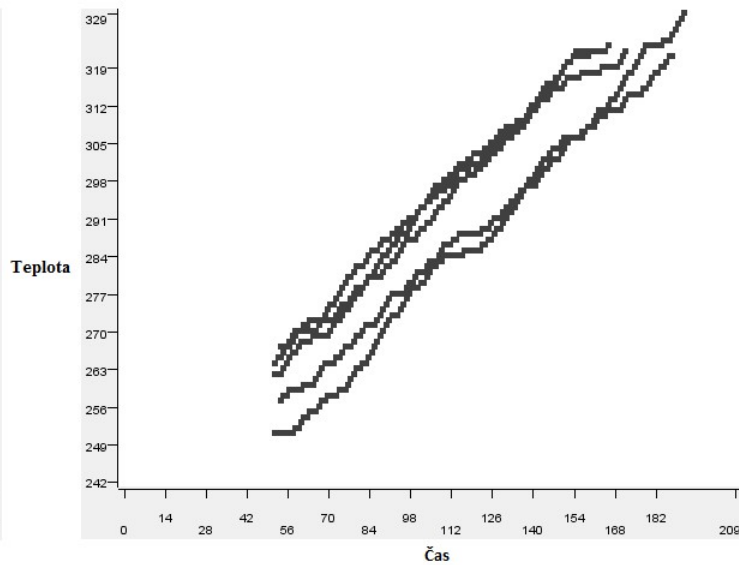
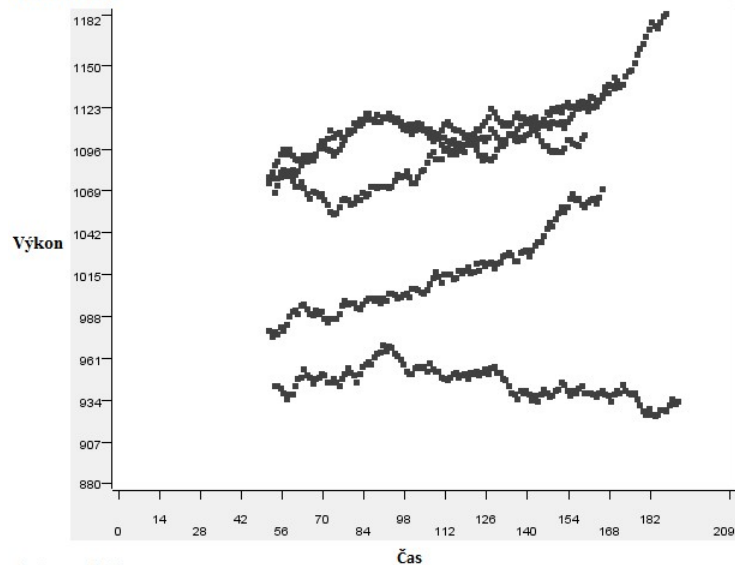
Obrázek 4: závislost výkonu na čase

U každého chybového stavu, který jsem za pomoci uzlu *Row Filter* vyseparoval, budu nyní chtít zjistit závislosti, které se v časové řadě vyskytují. Z analýzy grafů je pak zřejmé, že při chybě 101 je teplota s časem konstantní, ale výkon s časem klesá. Při chybě 202 však teplota s časem roste a výkon osciluje. U poslední chyby s kódem 303 teplota je s časem konstantní a výkon s časem klesá. Z grafů (Obrázek 5: závislost atributu výkon a teplota na atributu *aktualniStav*) jsem zjistil, že změna teploty a výkonu jsou podstatnými atributy pro budoucí predikci.

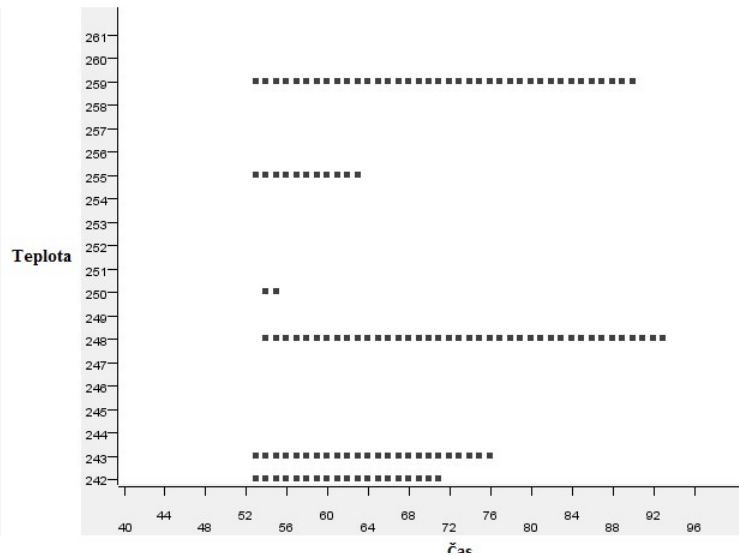
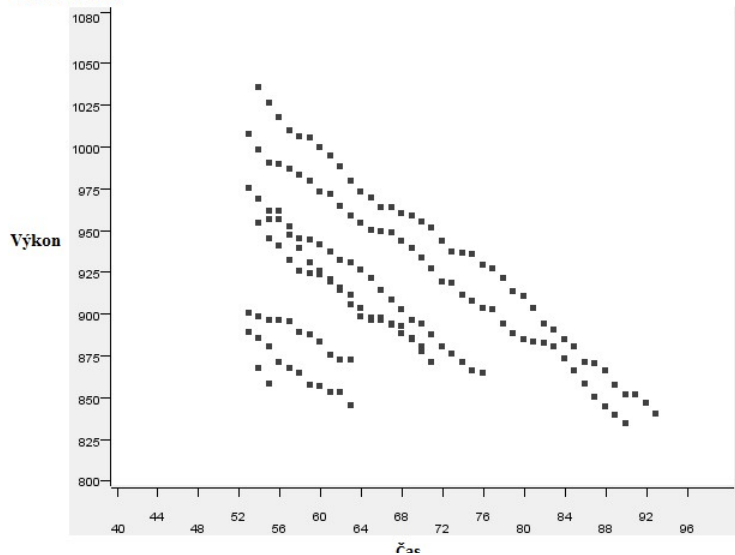
Outcome 101



Outcome 202



Outcome 303



Obrázek 5: závislost atributu výkon a teplota na atributu *aktualniStav*

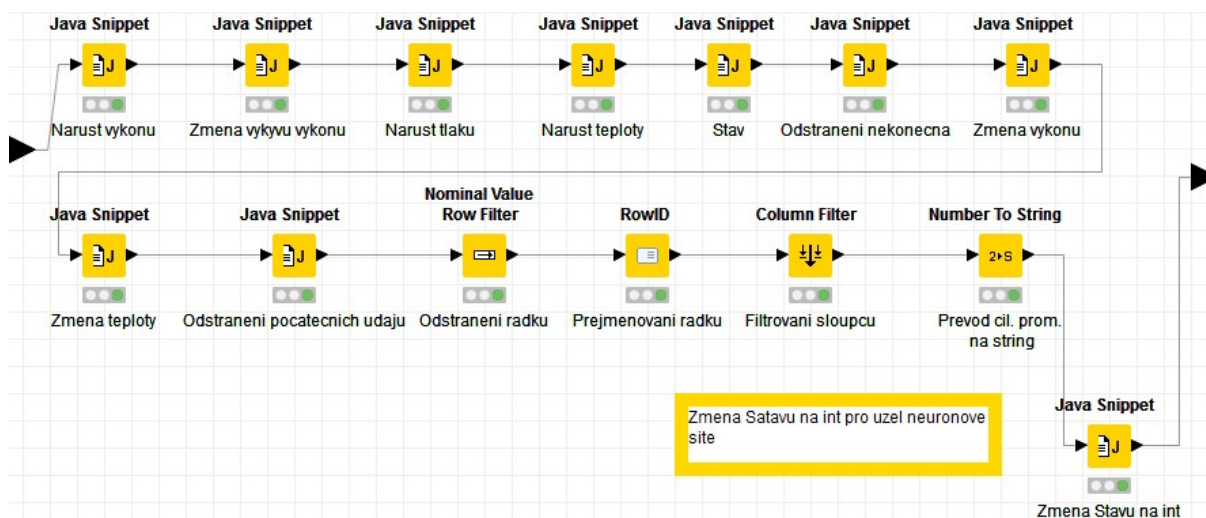
Samozřejmě by nás dále mohlo zajímat, kolik máme časových řad k dispozici, jejich velikost, v kolika časových řadách došlo k chybám apod. V příloze (*Monitorování provozu stroje - zkoumání dat*) je k dispozici několik otázek, kterými jsem analyzoval datový soubor použitý v tomto modelu a následně to vedlo k porozumění datům.

4. Příprava dat

Při tvorbě data miningového modelu je v prvním kroku potřeba připravit data před dalším použitím. Příprava dat je důležitým krokem před tvorbou modelů či dalším zkoumáním.

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. Za tímto účelem byl vytvořen tzv. **Meta node**. *Meta node* slouží k zapouzdření zvolených uzlů do uzlu jednoho. Tento uzel lze přirovnat k metodě (funkci) v programovacích jazycích. Uzel lze vytvořit následujícím způsobem: v kontextové nabídce stiskneme záložku *Node->Open Meta Node Wizard...* Následně se nám otevře průvodce pro vytvoření tohoto uzlu.

V tomto průvodci lze zvolit počet vstupů a výstupů spolu s názvem uzlu. Pro naše účely byl vytvořen uzel s názvem **Příprava dat**, který obsahuje jeden vstup a jeden výstup. Na následujícím obrázku je zobrazena struktura uzlů, která byla vytvořena v tomto uzlu.

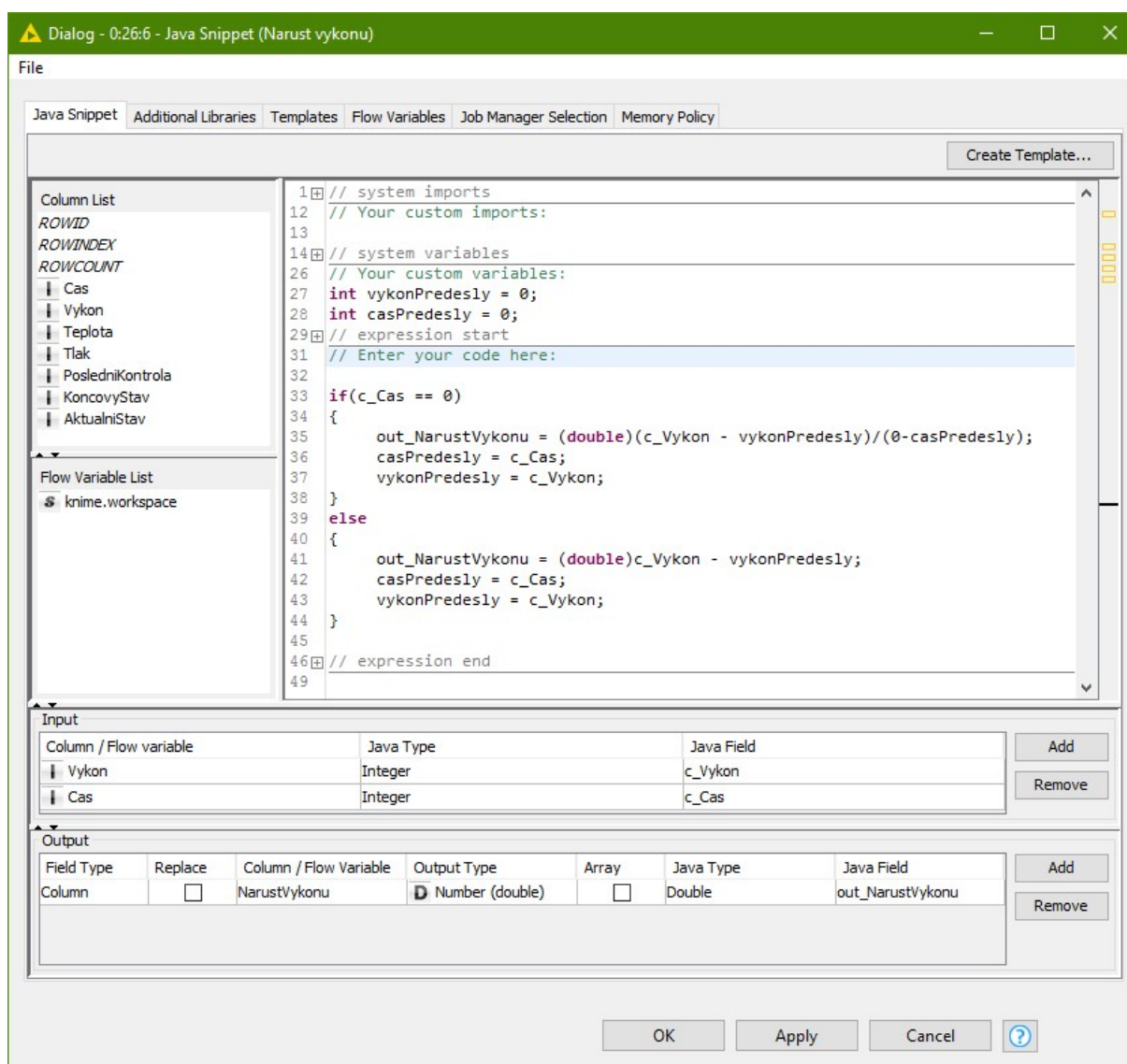


Obrázek 6: uzel *Příprava dat*

V několika následujících bodech jsou popsány nově odvozené uzly.

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

Pro zjištění rozdílu ve dvou po sobě následujících záznamech jsem použil uzel **JAVA Snippet**, pomocí kterého jsem požadované chování naprogramoval. Důvodem této volby bylo, že jsem nenašel uzel, který by tímto chováním disponoval. Ukázku zdrojového kódu uzlu *zmena výkonu* je zobrazena na *Obrázek 7: uzel pro zjištění zmeny výkonu*.



Obrázek 7: uzel pro zjištění zmeny výkonu

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

- **Pocet vykvyvu vykonu** – vrací hodnotu *TRUE* pokud jsou dva následující záznamy výkonu opačné (liší se znaménka). *Obrázek 8: kód Java* zobrazuje kód jazyka Java, který vykonává tuto činnost.

```
double powerIncPredesly = 0;
// expression start
public void snippet() throws TypeException, ColumnException, Abort {
// Enter your code here:

if((c_Time>0) && (c_PowerInc * powerIncPredesly <0))
    out_PowerFlux = true;
else
    out_PowerFlux = false;

powerIncPredesly = c_PowerInc;
// expression end
}
```

Obrázek 8: kód Java uzlu *Pocet vykvyvu vykonu*

- **Narust tlaku** – počet výstražných tlaků pro každou časovou řadu zvlášť. *Obrázek 9: kód Java uzlu Narust tlaku* zobrazuje kód tohoto uzlu v programovacím jazyce Java.

```
int counter = 0;
// expression start
public void snippet() throws TypeException, ColumnException, Abort {
// Enter your code here:

if(c_Pressure!=0)
    counter++;
if(c_Time == 0)
    counter = 0;

out_PressureWarnings = counter;

// expression end
}
```

Obrázek 9: kód Java uzlu *Narust tlaku*

Pro nárůst výstražných tlaků se ukládá jejich počet v časové řadě do aktuálního záznamu. Situace je zachycena na obrázku *Obrázek 10: nárůst výstražných tlaků*

Row ID	Cas	Vykon	Teplota	Tlak	NarustTlaku
Row736	68	888	252	1	1
Row737	69	887	252	0	1
Row738	70	887	252	1	2
Row739	71	887	252	0	2
Row740	72	885	252	0	2
Row741	73	885	252	0	2
Row742	74	885	252	0	2
Row743	75	882	252	0	2
Row744	76	878	252	0	2
Row745	77	877	252	0	2
Row746	78	873	252	0	2
Row747	79	870	252	0	2
Row748	80	870	252	0	2
Row749	81	870	252	0	2
Row750	82	868	252	1	3
Row751	83	867	252	1	4

Obrázek 10: nárůst výstražných tlaků

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

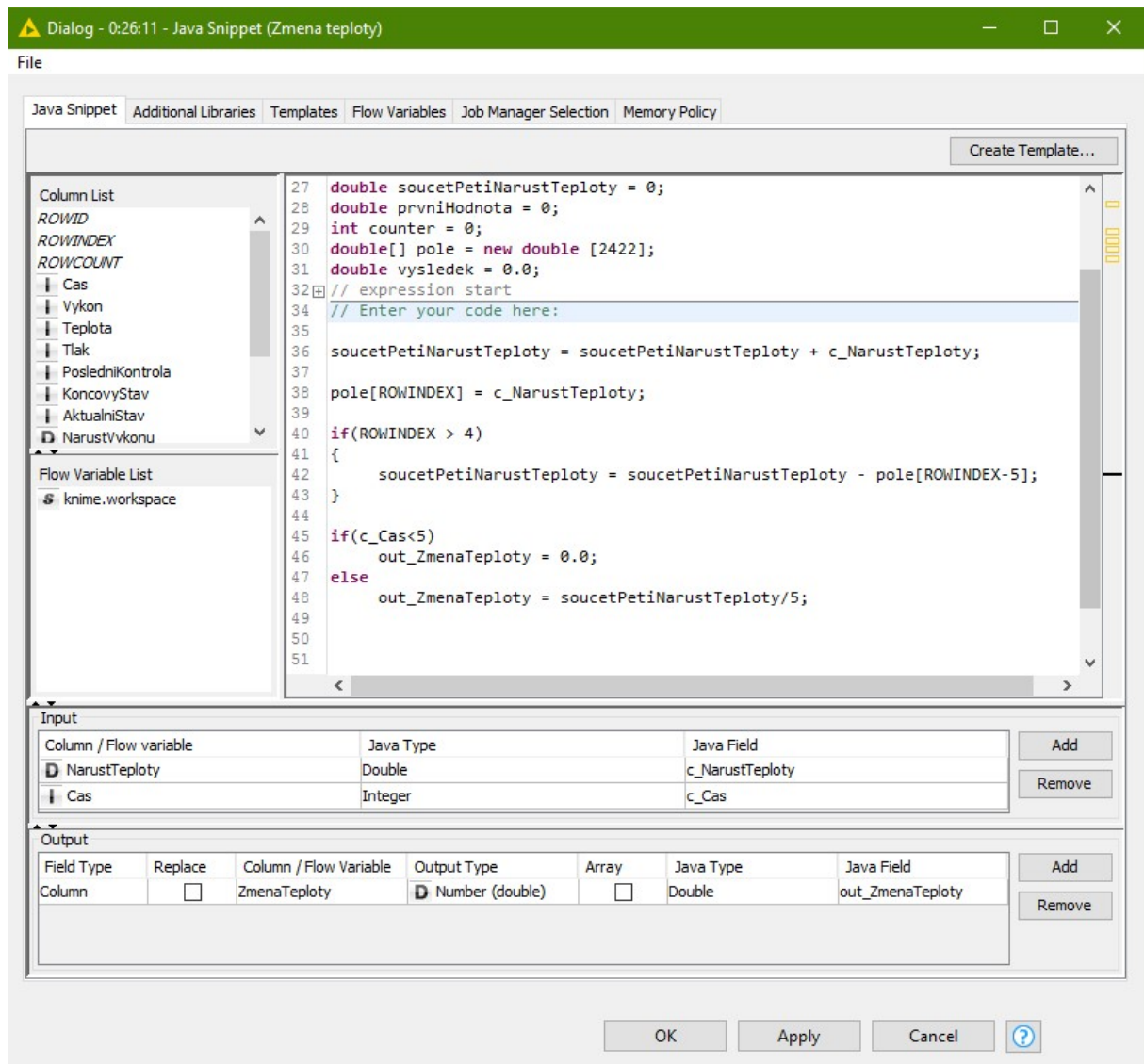
Pro uzel výpočet nárůst tlaku nebyla již využita stejná statistická metoda klouzavého průměru a to z důvodu, že tlak je reprezentován hodnotou 1 a 0 a nechceme zkreslovat data, chceme znát výkyv hodnot, proto s tlakem pracujeme jinak.

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 *změna teploty* a *změna výkonu*. Uzly používají metodu klouzavého průměru pro vyhlazení možných výkyvů hodnot způsobené chybou měření či anomálií, které mohly nastat. Ve výchozím nastavení byl klouzavý průměr nastaven na hodnotu 5. Pro odvození atributů popisující změnu teploty a výkonu byl použit uzel *JAVA Snippet*, ve kterém bylo toto chování naprogramováno. Programový kód uzlů je demonstrován na změně výkonu viz. *Obrázek 11: uzel změna teploty*.

V následujících bodech jsou popsány nově odvozené uzly:

- *Změna výkonu* - klouzavý průměr hodnot, které produkuje uzel *Narůst výkonu*
- *Změna teploty* – klouzavý průměr hodnot, které produkuje uzel *Narůst teploty*

- **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*.



Obrázek 11: uzel *zmena teploty*

Stejně tak jako volba klouzavého průměru, nastavení hodnoty v uzlu *Stav*, po které uzel přechází zpět do stabilního stavu, ovlivňuje přesnost modelu. Klouzavý průměr byl ve výchozím nastavení nastaven na hodnotu 5. Testováním různých konfigurací se zabývám v diplomové práci.

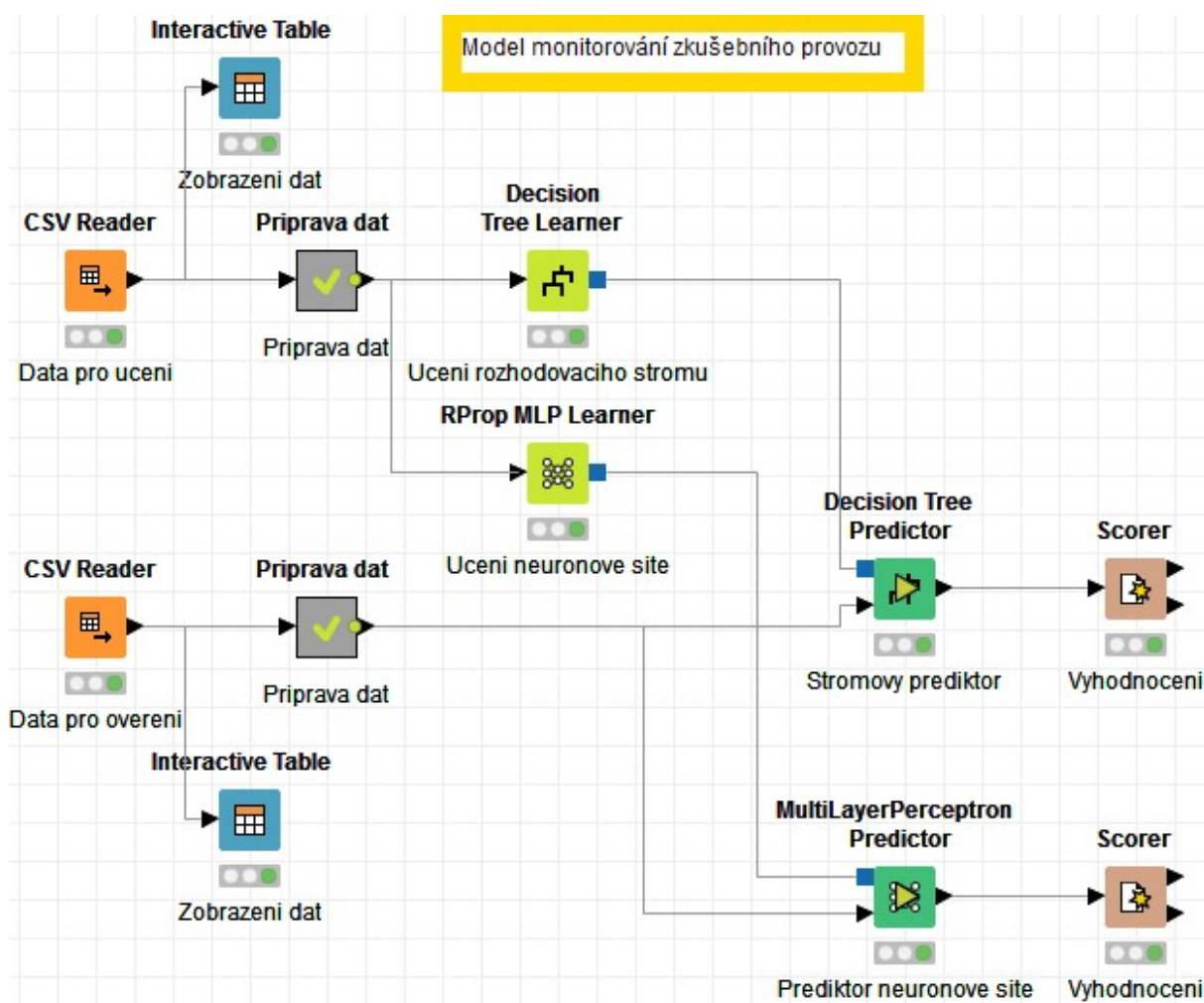
Zbývající část přípravy dat obsahuje několik dalších vytvořených uzlů. Jedním z nich je uzel pro filtrování dále nepotřebných atributů z datové matice, protože jsou zakompovány do jiných uzlů a pro další zpracování již nejsou potřeba. Mezi tyto atributy patří *Pocet vyskytu vykonu*, který je využit v 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*.

V následujících bodech jsou popsány zbylé uzly použití v uzlu *Priprava dat*:

- ***Odstraneni nekonecna*** – slouží k odstranění hodnoty nekonečna, která se vytvoří v uzlech *Narust teploty* a *Narust vykonu*. Tyto hodnoty se zobrazí z důvodu, že uzly počítají s hodnotami předcházejících a současných záznamů. Jelikož předešlé hodnoty neexistují, nekonečno se zobrazí v prvních řádcích zmíněných uzlů.
- ***Odstraneni pocatecnich udaju*** – slouží k označení prvního řádku každé řady hodnotou *TRUE*. Tyto řádky budou později odebrány, protože obsahují nedefinované odvozené diferenční charakteristiky.
- ***Odstraneni radku*** – uzel, který odstraňuje předešle označené řádky
- ***Prejmenovani radku*** – po odstranění konkrétních řádku se názvy jednotlivých řádku sami nepřeindexují. Z tohoto důvodu je nutné tyto řádky přeznačit, aby na sebe řádky navazovaly.
- ***Filtrovani sloupcu*** – uzel pro odstranění sloupců, které nebudeme v modelování potřebovat
- ***Prevod cil.prom. na string*** – převod proměnné *aktualniStav* na typ *string*. Tento převod je uskutečněn pro uzly, které používám pro modelování. Tyto uzly pro predikování vývoje konkrétní proměnné požadují proměnnou typu *string*.
- ***Změna powerState na int*** – tento uzel převádí stavy *Stable* a *Fluctuating* na hodnoty 1 a 0 z důvodu, že uzly pro predikci požadují proměnné, které tento uzel používá k predikci cílové proměnné, aby byly číselné hodnoty.

5. Monitorování provozu

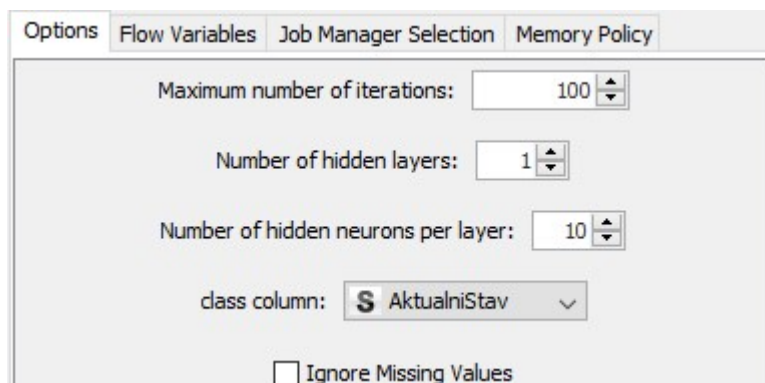
Model monitorování poruchy strojů se skládá z již zmíněné části přípravy dat a také z rozhodovacích algoritmů a to z rozhodovacího stromu a neuronové sítě. Model je aplikován jak na testovací data, tak i na data pro „učení“ pro transformaci vstupů do stejné struktury. Poslední částí modelu je uzel reprezentující *matici záměn* pro vyhodnocení úspěšnosti naučeného modelu. V modelu je dále připojen datový soubor *monitorovaniTestovani*, na kterém je prováděno testování naučeného modelu. Pro model jsem vytvořil *workflow Monitorovani_model*. Workflow je zobrazen na následujícím obrázku.



Obrázek 12: *workflow Monitorovani_model*

Stejně tak jako v předchozí části, ve které jsme se zabývali zobrazením závislostí času na jiných veličinách, jsou zde uzly pro načtení dat a to jak pro data (soubor *monitorovaniUceni*), které budou použita pro natrénování modelu, tak i pro testovací data (soubor *monitorovaniUceni*) k ověření námi vytvořeného modelu. V modelu je obsažen uzel *Příprava dat*, který byl popsán v minulé kapitole, a slouží k úpravě dat do požadované formy.

Nyní máme vše připraveno pro samotnou predikci dat, kde budeme využívat dva rozhodovací algoritmy. Jedná se o algoritmy rozhodovací strom a neuronová síť. Dalšími uzly ve *workflow Monitorovani_learn* tvoří uzly **Decision Tree Learner** a **RProp MLP Learner**. Uzly slouží pro učení rozhodovacího stromu a neuronové sítě. Nastavení těchto uzlů lze vidět na následujících obrázcích.



Options Flow Variables Job Manager Selection Memory Policy

Maximum number of iterations: 100

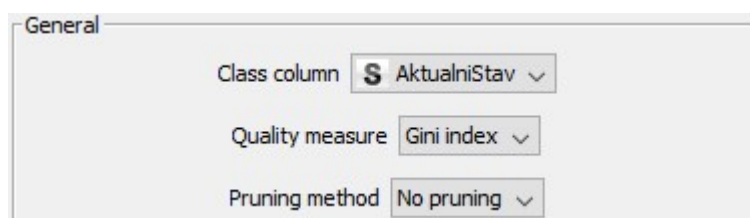
Number of hidden layers: 1

Number of hidden neurons per layer: 10

class column: S AktualniStav

☐ Ignore Missing Values

Obrázek 13: nastavení rozhodovacího stromu



General

Class column: S AktualniStav

Quality measure: Gini index

Pruning method: No pruning

Obrázek 14: nastavení neuronové sítě

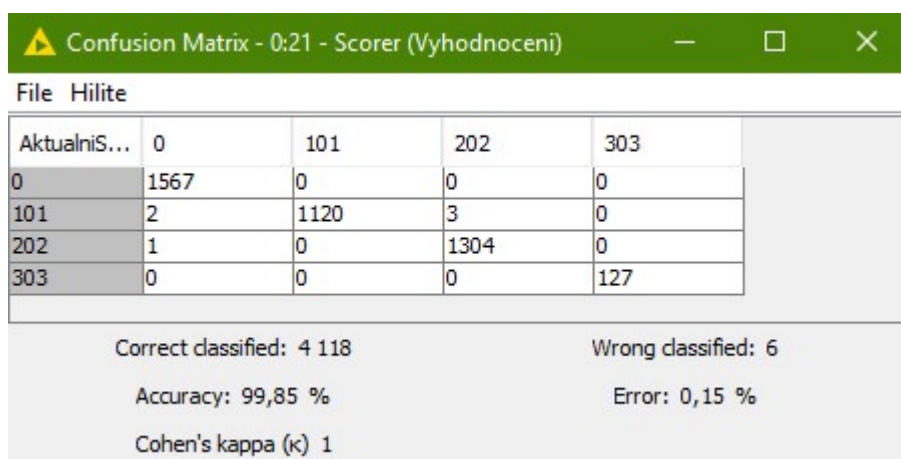
Dalšími uzly jsou samotné *prediktory*. Do *prediktoru* vstupují data z naučených uzlů a testovací data, která byla upravena stejným způsobem jako data pro učení, tedy byla upravena uzlem *Příprava dat*.

Za *prediktory* byly zapojeny uzly pro vyhodnocování výsledků (uzel **Scorer**). *Scorer* obsahuje **matici záměn** (confusion matrix), která slouží k vyhodnocení klasifikační úlohy. Určuje, zda byly predikované hodnoty správně zařazeny do tříd (v našem případě rozdělení do chybových stavů). Skutečnost správného zařazení nám zobrazuje diagonála matice. Hodnoty mimo diagonálu značí špatné zařazení *prediktorem*.

6. Výsledky

Naučení a samotné predikce modelu dopadly velmi dobře. Pro rozhodovací strom se *prediktor* spletl pouze v šesti případech a to v chybovém stavu 101. Výsledná přesnost tohoto *prediktoru* byla 99,85%. Druhý *prediktor* neuronové sítě zaznamenal horší výsledky. Jeho celková přesnost byla 98,23% a spletl se v 73 případech. Zajímavostí je, že *prediktor* mylně zařazoval chybový stav s číslem 202 jako stav bezchybný.

Dále je nutné zmínit, že neuronová síť při každém novém spuštění vykazovala jiné hodnoty přesnosti a je to zaviněno inicializací vlastní neuronové sítě, která využívá k inicializaci náhodné číslo. Takovéto chování je možné potlačit zaškrtnutím volby *Use seed for random initialization* a zvolit inicializační číslo ručně. Inicializačním číslem budeme dostávat stejné výsledky s každým novým spuštěním predikce. V této úloze jsem zvolil inicializační číslo na hodnotu 284 681 600. Výsledky jednotlivých *matic záměn* jsou zobrazeny na následujícím obrázku.



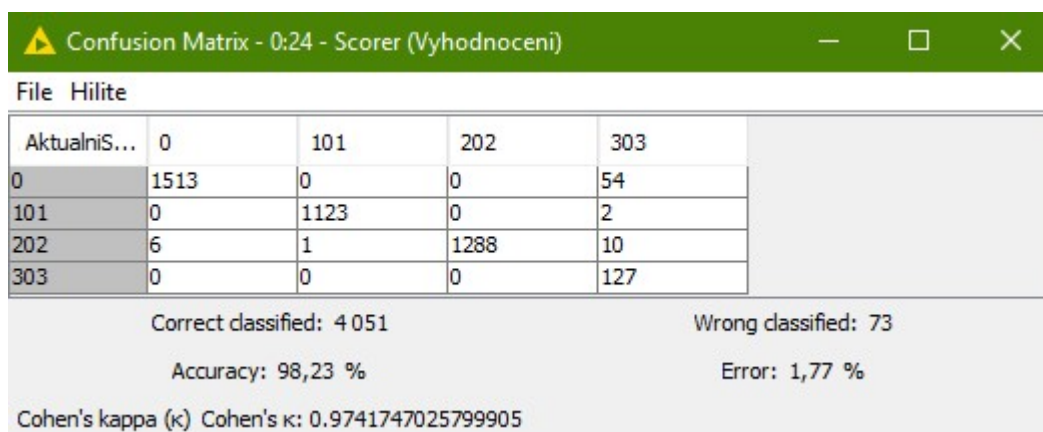
Confusion Matrix - 0:21 - Scorer (Vyhodnoceni)

File Hilite

AktualniS...	0	101	202	303
0	1567	0	0	0
101	2	1120	3	0
202	1	0	1304	0
303	0	0	0	127

Correct classified: 4 118 Wrong classified: 6
Accuracy: 99,85 % Error: 0,15 %
Cohen's kappa (κ) 1

Obrázek 15: matice záměn rozhodovacího stromu



Confusion Matrix - 0:24 - Scorer (Vyhodnoceni)

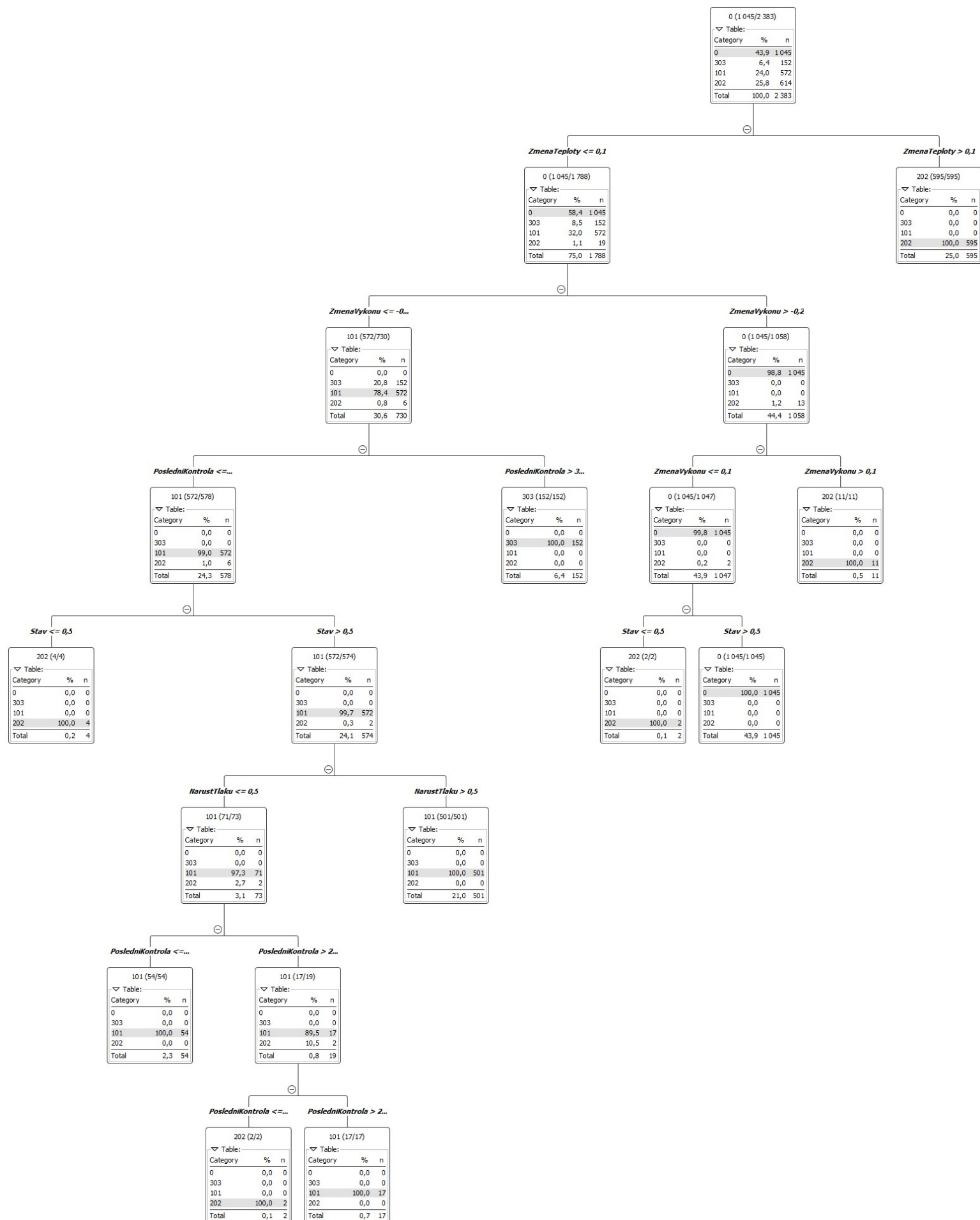
File Hilite

AktualniS...	0	101	202	303
0	1513	0	0	54
101	0	1123	0	2
202	6	1	1288	10
303	0	0	0	127

Correct classified: 4 051 Wrong classified: 73
Accuracy: 98,23 % Error: 1,77 %
Cohen's kappa (κ) Cohen's κ : 0.9741747025799905

Obrázek 16: matice záměn neuronové sítě

Rozhodovací strom dále nabízí možnost zobrazit průběh rozhodování, tzv. podle kterých kritérií se rozhodoval. Zobrazení tohoto stromu je možné zobrazit pravým kliknutím na *prediktor* a vybráním možnosti **View: Decision Tree View**. Tím se nám zobrazí celý rozhodovací strom. V tomto stromu (*Obrázek 17: rozhodovací strom*) například vidíme, že nejdůležitějším parametrem pro rozhodování byl atribut *zmena teploty*, který udává, že pokud nastává růst teploty, tak se stroj dostane do chybového stavu 202. Dalším významným atributem pro rozhodování byla změna výkonu stroje. Rozhodovací strom lze tedy tímto způsobem jednoduše interpretovat. Neuronová síť však tímto způsobem zobrazit nelze.



Obrázek 17: rozhodovací strom

7. 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 *Obrázek 17: rozhodovací strom*. 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, tzv. segmentace 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.

Seznam obrázků

Obrázek 1: přehled vstupních dat	2
Obrázek 2: zobrazení závislostí sledovaných veličin	4
Obrázek 3: nastavení uzlu row filter	5
Obrázek 4: závislost výkonu na čase	6
Obrázek 5: závislost atributu výkon a teplota na atributu <i>aktualniStav</i>	7
Obrázek 6: uzel <i>Priprava dat</i>	8
Obrázek 7: uzel pro zjištění <i>zmeny vykonu</i>	9
Obrázek 8: kód Java uzlu <i>Pocet vykvyu vykonu</i>	10
Obrázek 9: kód Java uzlu <i>Narust tlaku</i>	10
Obrázek 10: nárůst výstražných tlaků	11
Obrázek 11: uzel <i>zmena teploty</i>	12
Obrázek 12: <i>workflow Monitorovani_model</i>	14
Obrázek 13: nastavení rozhodovacího stromu	15
Obrázek 14: nastavení neuronové sítě	15
Obrázek 15: matice záměn rozhodovacího stromu	16
Obrázek 16: matice záměn neuronové sítě	16
Obrázek 17: část stromu rozhodovacího stromu	18

Seznam tabulek

Tabulka 1 přehled atributů vstupního souboru	2
--	---