

UNIVERZITA PARDUBICE
FAKULTA EKONOMICKO-SPRÁVNÍ

DIPLOMOVÁ PRÁCE

2011

Bc. Petr ŠIMER

Univerzita Pardubice
Fakulta ekonomicko-správní

Predikce akciových kurzů pomocí posilovaného učení

Bc. Petr Šimer

Diplomová práce
2011

Univerzita Pardubice
Fakulta ekonomicko-správní
Akademický rok: 2010/2011

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)


Jméno a příjmení: **Bc. Petr ŠIMER**
Osobní číslo: **E09836**
Studijní program: **N6209 Systémové inženýrství a informatika**
Studijní obor: **Informatika ve veřejné správě**
Název tématu: **Predikce akciových kurzů pomocí posilovaného učení**
Zadávající katedra: **Ústav systémového inženýrství a informatiky**


Z á s a d y p r o v y p r a c o v á n í :

Současný stav predikce akciových kurzů.
Charakteristika posilovaného učení.
Předzpracování finanční časové řady.
Návrh modelu pro predikci akciových kurzů.
Zhodnocení výsledků a porovnání s výsledky metod učení s učitelem.
Vypracování uživatelské příručky.

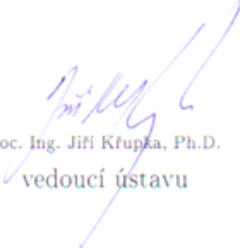
Rozsah grafických prací:
Rozsah pracovní zprávy: cca 60 stran
Forma zpracování diplomové práce: tištěná/elektronická
Seznam odborné literatury:

- KAELBLING, L. P., LITTMAN, M. L., MOORE, A. W. Reinforcement Learning : A Survey. Journal of Artificial Intelligence Research. 1996, vol. 4, pp.237-285.
LEE, J. W. Stock Price Prediction Using Reinforcement Learning. IEEE International Joint Conference on Neural Networks. 2001, pp.690-695.
OLEJ, V. Modelovanie ekonomických procesov na báze výpočtovej inteligencie. Hradec Králové : Miloš Vognar-MandV, 2003. 160 s., ISBN 80-903024-9-1.
SCHÖNEBURG, E. Stock Price Prediction using Neural Networks: A Project Report. Neurocomputing. 1990, vol. 2, no. 1, pp.17-27.
SUTTON, R. S., BARTO, A. G. Reinforcement Learning. Cambridge : MIT Press, 1998.

Vedoucí diplomové práce: 
Ing. Petr Hájek, Ph.D.
Ústav systémového inženýrství a informatiky
Datum zadání diplomové práce: 5. října 2010
Termín odevzdání diplomové práce: 6. května 2011


doc. Ing. Renáta Myšková, Ph.D.
děkanka

L.S.


doc. Ing. Jiří Křupka, Ph.D.
vedoucí ústavu

V Pardubicích dne 5. října 2010

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 20. 6. 2011

Petr Šimer

PODĚKOVÁNÍ

Tímto bych rád poděkoval vedoucímu diplomové práce, panu Ing. Petru Hájkovi, Ph.D. za odbornou pomoc, cenné rady a připomínky k obsahové i formální stránce, za čas, který mi věnoval a za vedení a podporu při vypracování diplomové práce.

ANOTACE

Práce se zabývá predikcí akciových kurzů pomocí metody posilovaného učení. V úvodu je popsán současný stav predikce akciových kurzů včetně základních pojmů a akciových analýz. Následně jsou vysvětleny důležité pojmy z oblasti posilovaného učení. Dále se podrobně věnuji jednotlivým součástem a elementárním metodám řešení posilovaného učení. Předzpracování a predikce finanční časové řady jsou provedeny pomocí algoritmu genetického posilovaného učení za použití open source java softwarového nástroje v prostředí produktu Keel. V závěru jsou vyhodnoceny výstupy a interpretovány výsledky, které jsou porovnány s výsledky metod učení s učitelem a vypracována uživatelská příručka.

KLÍČOVÁ SLOVA

posilované učení, predikce akciových kurzů, genetické algoritmy, algoritmus genetického posilovaného učení pomocí symbiotické evoluce, předzpracování dat, finanční časové řady, učení s učitelem

TITLE

Stock prices prediction by reinforcement learning

ANNOTATION

This work deals with stock price prediction using reinforcement learning method. In the introduction is described the current state of stock price prediction including the basic terms and stock analyses. Consequently, the important terms of reinforcement learning are explained. Further on, I deal in detail with particular items and elementary methods of reinforcement learning. Preprocessing and prediction of financial time series is done using genetic reinforcement learning algorithm within open source java software tool in the environment of Keel software product. Finally, the outputs are assessed and the results are interpreted and compared with the results of supervised learning methods. Also, a user manual is created.

KEYWORDS

reinforcement learning, stock price prediction, genetic algorithm, genetic reinforcement learning algorithm through symbiotic evolution, data preprocessing, financial time series, supervised learning

Obsah

Úvod	9
1 Současný stav predikce akciových kurzů	10
1.1 Úvod do časových řad	10
1.2 Akcie	10
1.3 Akciové trhy a akciové indexy	11
1.4 Akciové analýzy	13
1.4.1 Fundamentální analýza	14
1.4.2 Technická analýza	15
1.4.3 Psychologická analýza	18
1.5 Dílčí závěr	20
2 Posilované učení	21
2.1 Historie posilovaného učení	21
2.2 Markovský rozhodovací proces	23
2.3 Základní pojmy posilovaného učení	25
2.3.1 Rozhraní agent - prostředí	25
2.3.2 Cíle a odměny	27
2.3.3 Výnosy	28
2.3.4 Hodnotové funkce	29
2.3.5 Optimální hodnotové funkce	30
2.4 Elementární metody řešení	31
2.4.1 Dynamické programování	31
2.4.2 Monte Carlo metody	33
2.4.3 Temporal Difference učení	34
2.4.4 Fuzzy systémy adaptované genetickými algoritmy a symbiotickou evolucí	35
2.4.4.1 Kódování, inicializace a vhodnost přiřazení	36
2.4.4.2 Reprodukce, křížení a mutace	39
2.5 Dílčí závěr	41
3 Předzpracování finanční časové řady	42
3.1 Zdrojová data	42
3.2 Předzpracování časové řady	43
3.2.1 Jednoduchý klouzavý průměr	44
3.2.2 Vážený klouzavý průměr	45
3.2.3 Indikátor momentum	46
3.2.4 Směrodatná odchylka	47
3.3 Dílčí závěr	48

4	Návrh modelu pro predikci akciových kurzů	49
4.1	Návrhu modelu	49
4.2	Transformace dat	50
4.3	Rozdělení dat na trénovací a testovací množinu	51
4.4	Algoritmus fuzzy symbiotické evoluce	52
4.5	Dílčí závěr	53
5	Zhodnocení výsledků experimentů	54
5.1	Nastavení parametrů algoritmu	54
5.2	Zhodnocení výsledků algoritmu	54
5.3	Porovnání s výsledky metod učení s učitelem	57
5.3.1	Neuronová síť RBF	57
5.3.2	Dopředná neuronová síť typu perceptron	59
5.3.3	Porovnání výsledků predikce	60
6	Uživatelská příručka Keel 2.0	62
6.1	Příprava dat	63
6.2	Návrh experimentu	70
6.3	Složky panelu experimentu	72
6.4	Spuštění experimentu	76
	Závěr	78
	Seznam použité literatury	80
	Seznam obrázků	83
	Seznam grafů	84
	Seznam tabulek	85
	Seznam příloh	86
	Přílohy	87

Úvod

Představa, že se učíme prostřednictvím interakce s prostředím, je pravděpodobně první, která nás napadne, přemýšlíme-li o podstatě učení. Když si dítě hraje, mává svými pažemi, nebo se dívá kolem sebe, není zde žádný explicitní učitel, ale je zde přímé senzomotorické propojení se svým prostředím. Uplatnění tohoto propojení produkuje obrovské množství informace o příčině a následku, o důsledcích akce, a o tom, co dělat pro dosažení cílů. Po celý život, jsou takové interakce nepochybně hlavním zdrojem poznatků o prostředí člověka a o něm samotném. Ať už se člověk učí řídit auto nebo vést rozhovor, tak si plně uvědomuje, jak toto prostředí reaguje na to, co dělá, a snaží se ovlivňovat, co se stane prostřednictvím chování. Učení se z interakce je základním myšlenkou téměř všech teorií učení a inteligence, včetně posilovaného učení.

Cílem této práce je provést predikci cen akcií pomocí posilovaného učení. K tomuto účelu bude navržen model pro predikci cen akcií, který bude realizován na předzpracované časové řadě. Výsledky experimentů budou porovnány s výsledky dosaženými pomocí dalších metod učení s učitelem.

Práce je tematicky rozdělena na několik kapitol. První kapitola je zaměřena na všeobecný popis současného stavu predikce akciových kurzů, vysvětlení základních pojmů a metod predikce. Po přiblížení těchto metod jsou dále popsány metody posilovaného učení. Další kapitola je věnována předzpracování finanční časové řady pro predikci akciových kurzů. Jako vstupní data je použita finanční časová řada o délce 4 roky obsahující hodnoty akciového indexu Nasdaq Composite, který se stal vůbec nejsledovanějším díky svému širokému záběru měřicímu vývoj všech akcií obchodovaných na The Nasdaq Stock Market, a to jak amerických, tak i neamerických. Jeho komponenty je v současné době přes tři tisíce akciových titulů. Předzpracování této časové řady je založeno na použití metod technické analýzy akcií. Další kapitola obsahuje návrh modelu pro predikci. Potom následuje vyhodnocení výstupů a interpretace dosažených výsledků v porovnání s metodami učení s učitelem. Práce obsahuje také uživatelskou příručku použitého programového prostředí Keel.

1 Současný stav predikce akciových kurzů

Cílem predikce cen akcií je získat nástroj na podporu rozhodování o investicích. Byly vyvinuty různé metody založené na fundamentální analýze, technických indikátorech nebo psychologické analýze akcií. Pro predikci cen akcií se používají nejčastěji indikátory technické analýzy akcií. Zvláště pomocí systémů využívajících metod umělé inteligence bylo dosaženo lepších výsledků ve srovnání s konvenčními metodami.

1.1 Úvod do časových řad

Akciové kurzy [21] jsou reprezentovány časovými řadami. Predikce časových řad je problematika, která se dotýká mnoha oborů, od statistiky po ekonomické obory. V rámci predikce časových řad je cílem předpovědět budoucí hodnoty, nalézt mechanismy, na nichž jsou závislé hodnoty časové řady a faktory, které je ovlivňují. Pod pojmem časová řada si lze představit sekvenci hodnot, jež je závislá na čase. Příkladem jsou například počty narozených dětí v daných dnech, množství vyrobených produktů v podniku či cena určitého titulu na daném akciovém trhu.

Časová řada se může skládat ze dvou složek. Těmi jsou složka deterministická a stochastická. Deterministickou složku lze dělit dále na složky trendové, složky s konstantní nebo nekonstantní periodou, apod. Samotnou časovou řadu tak lze rozložit na několik složek a každá může mít z pohledu predikce svou váhu [21].

Pro predikci časových řad se využívá mnoho metod, jak z oblasti statistiky, tak především z oblasti umělé inteligence či soft computingu, např. neuronové sítě, fuzzy logika atd. V rámci různých optimalizačních problémů spojených s predikcí časové řady je také možno použít evolučních algoritmů. Při predikci časové řady hrají klíčový význam metody využití k predikci a znalost složek časové řady. Při predikci je vhodné vyhodnocovat chyby předpovědi a zlepšovat podle těchto chyb výsledný model [21].

1.2 Akcie

Akcie je cenný papír, který zakládá svému majiteli (akcionáři) podíl na majetku akciové společnosti. Práva a povinnosti majitele akcie upravuje kromě obchodního zákoníku také zákon o cenných papírech [10].

Akcionář má mimo jiné právo na podíl na zisku akciové společnosti - na tzv. dividendu. O její výši rozhoduje každý rok s ohledem na dosažený hospodářský výsledek valná hromada složená z akcionářů. Pokud akciová společnost docílí zisku, převyšuje zpravidla dividendový výnos zhodnocení instrumentů s pevným výnosem [10].

Častěji než na dividendu upírají majitelé akcií pozornost na vývoj kurzu akcie na burzách cenných papírů. Růst kurzu jim může přinést kromě dividend další zhodnocení investice. V historickém pohledu převyšuje výnos z akcií zhodnocení alternativních typů investic (obligace, termínované vklady, nemovitosti apod.) [10].

Existují následující druhy akcií [1]:

- na majitele – převoditelná pouhým předáním, práva z akcie náleží majiteli.
- na jméno – převoditelná rubopisem a každý nový majitel je zapsán v seznamu akcionářů.
- zaměstnanecké – majiteli jsou zaměstnanci a držitelé těchto akcií jsou zvýhodněni, např. mohou si koupit akcii za polovinu nominální hodnoty.
- prioritní – držitelé mají určité výhody, mají přednostní právo na vyplácení dividend.
- zlaté – zde jsou vlastníky státní orgány např. ministerstva.

Akcie mohou mít podobu listinnou (akcie je samostatný list) nebo zaknihovanou (jsou zapsané ve středisku cenných papírů)

1.3 Akciové trhy a akciové indexy

Akciovým trhem se rozumí burzovní veřejný sekundární organizovaný finanční trh, na kterém ve vyspělých zemích obchoduje i veřejnost (na rozdíl od burz devizových, na nichž obchodují především banky) [21]. Veřejným trhem se rozumí trh, na kterém mohou obchodovat všichni potencionální zájemci a dané předmětné tituly jsou obchodovány za nejvyšší nabídnuté ceny. Pod pojmem sekundární trh se rozumí trh, na kterém se obchoduje s cennými papíry, které již byly v minulosti vydány, na rozdíl od trhu primárního, na kterém dochází právě k emisi nových cenných papírů. Pod pojmem organizovaný trh je chápán trh, který je organizován jistou institucí, konkrétně u burzovního organizovaného trhu je touto institucí burza, která má za úkol agregovat veřejnou nabídku a poptávku a párovat dané obchodní příkazy. Obecně se dá říci, že čím významnější jednotlivé burzy jsou (a s nimi spojený trh), tím kvalitnější a světově známější obchodní tituly se na těchto trzích obchodují [21].

Každý akciový trh má jistá specifika a svou historii. Především jej charakterizuje fluktuační úroveň jeho cen, vztah mezi cenami jednotlivých akcií a celku, jejich zisky a dividendami. Investor, který se chce účastnit obchodování na daném akciovém trhu, by měl znát jeho charakteristiky, vliv hospodářského cyklu na vývoj trhu atd. Přestože platí, že celý finanční trh lze členit podle několika hledisek na různé dílčí trhy, přesto se tyto části vzájemně ovlivňují a existují mezi nimi vazby. Mezi významný faktor integrující jednotlivé dílčí segmenty patří i spekulace. Právě spekulace, kterou představují spekulanti ovlivňující především tržní ceny obchodovaných titulů na daných trzích, jsou prvkem trhu, který je obtížně predikovatelný a je vysoce rizikový [21].

Mezi nejvýznamnější burzy patří především New York Stock Exchange (NYSE), která je považována za největší, nejlikvidnější a nejvýznamnější burzu cenných papírů. Kromě významných amerických společností jsou na této burze obchodovány také významné zahraniční tituly. V roce 2006 se tato burza spojila s burzovní aliancí Euronext a vytvořili tak nejsilnější uskupení. Reprezentativním indexem NYSE je Dow Jones Industrial Average. Další významnou burzou je Tokyo Stock Exchange (TSE), NASDAQ (další burza působící v Americe), Euronext (významné burzovní uskupení se sídlem v Paříži), London Stock Exchange (LSE), Shanghai Stock Exchange, Hong Kong Stock Exchange. Mezi významné indexy tak patří Dow Jones, S&P 500, NASDAQ a další.

Svůj vlastní index má téměř každá burza, existuje celá řada mimoburzovních indexů. Indexy jsou nejen základním statistickým indikátorem měřícím vývoj celého trhu, ale také určitým měřítkem úspěšnosti a schopností investora. Ten, který dokáže své investiční portfolio dlouhodobě zhodnocovat lépe, než činí přírůstek či ztráta daného indexu, tj. výnosnost portfolio převyšuje výnos celého trhu, je považován za úspěšného investora. Zde je uveden krátký přehled nejznámějších světových indexů [10]:

- *Dow Jones Industrial Average* se počítá již více než 100 let a považuje se za indikátor nálady na mezinárodních finančních trzích. Přitom jeho hodnota je dnes stále počítána podle mechanismu, který byl zaveden v roce 1896. Dow Jones je počítán jako cenově vyvážený index – všechny společnosti jsou v něm zastoupeny podle svých absolutních akciových kurzů. To způsobuje, že některé společnosti s vysokým akciovým kurzem mají v indexu významnou pozici, i když co se týče tržní kapitalizace, patří spíše k těm menším. Změny ve složení indexu jsou velice vzácné.
- *S&P 500 Index* se počítá od roku 1943 a obsahuje 500 titulů. Jednotlivé akcie se vybírají podle tržní hodnoty a celkového objemu obchodů. Další kritérium je oborová

reprezentativnost – váhové zastoupení by mělo zobrazovat aktuální oborovou strukturu amerického hospodářství.

- *NASDAQ 100 Index*, u něhož se váhové zastoupení upravuje čtvrtletně na bázi tržní kapitalizace. Díky vysoké likviditě nemají kritéria jako např. obchodovaný počet akcií nebo objemy obchodů význam. Jsou zde zastoupeny společnosti především z oblastí software, hardware, telekomunikace, infrastruktury sítí, informačně-technologických služeb nebo biotechnologie (Microsoft, Intel atd.).
- *S&P TSX 60 Index* je kanadský akciový index. Kritérii pro výběr akcií jednotlivých společností je vysoký objem burzovních obchodů a odpovídající tržní hodnota volně obchodovaných akcií.

1.4 Akciové analýzy

Základem úspěchu na akciových trzích je správné načasování nákupu a prodeje. Kurzy akcií jsou v neustálém pohybu a odhadnout jejich další vývoj bývá velice složité. Vývoj kurzu každé akcie je individuální. Přesto lze vyzorovat typický průběh kurzu a určité standardní situace, v nichž se kurz akcie může nacházet. Denní kurzy vytvářejí křivku, z níž lze vyčíst základní směr vývoje kurzu. Tento směr se nazývá trend. Rozlišuje se trend vzestupný, konstantní a sestupný. Většina investorů do akcií obchoduje s akciemi s respektem k trendu. Při stoupajících kurzech například investoři neustále nakupují, přičemž rostoucí poptávka po akciích vytlačuje kurzy akcií stále výše. V reakci na tuto situaci se v odborných kruzích říká, že "trend je tvůj přítel" [1].

Vzestupný trend zůstává zachován tak dlouho, dokud každé nové denní maximum kurzu neleží nad předchozím vrcholem. Jakmile každé nové minimum leží pod nejnižší hodnotou kurzu z předchozího dne, dochází k sestupnému trendu. Slabá poptávka po akciích se projevuje na poklesu kurzů akcií. K trendu patří jeho krátkodobé výchylky korigující předchozí přehnaný růst nebo výraznější pokles kurzu akcie [1].

Kurzy všech akcií se nevyvíjí stejným směrem. Protichůdný vývoj některých akcií může být způsoben ve fázi všeobecného růstu kurzů na burze například mimořádně chybnými rozhodnutími managementu akciových společností. Kurzy akcií takových společností klesají, přestože akcie společností působících ve stejném odvětví či zemi rostou. Naopak za situace, kdy většina akcií, se kterými se na burzách obchoduje, oslabuje, mohou některé akciové společnosti provést úspěšnou restrukturalizaci či zavedou krizový program cílený na zlepšení ziskovosti. Akcie těchto společností se stanou pro investory atraktivní. Vzroste po nich poptávka, která "vyžene" kurz akcie výše [1].

Klíč k úspěšnému obchodování na akciovém trhu je jednoduchý: ve správný okamžik akcie levně koupit a ve správný okamžik je draze se ziskem prodat. Problémem je, že kdy nastane ten správný okamžik, nikdo jistě neví. Proto vznikly akciové analýzy, které se na základě různých faktorů snaží nalézt ty, které tržní cenu akcií nějakým způsobem ovlivňují.

Cenový vývoj na akciovém trhu, jež je obecně vyjadřován indexy (v ČR PX50-BCPP a PK30-RMS), je jedním z nejdůležitějších předstihových ukazatelů v konjunkturním výzkumu pro indikaci hospodářských cyklů. Vývoj na akciovém trhu předbíhá skutečný vývoj HDP přibližně o 6 měsíců. Proto je analýza a predikce vývoje akcií důležitá pro pochopení a prognózy vývoje hospodářského produktu. Hlavní impuls pro změny akciových kurzů je vlastní zisk [1].

Lze říci, že se vznikem trhů s cennými papíry vznikly i první snahy o predikci budoucího vývoje kurzů. K rozmachu analytických metod došlo zejména po II. světové válce. To sebou přineslo i novou specializovanou profesi - akciový analytik. V bankách vznikly specializovaná oddělení pro analýzu cenných papírů a byly zakládány specializované analytické společnosti [1].

1.4.1 Fundamentální analýza

Fundamentální analýza se snaží předpovědět, jaké faktory působí a ovlivňují pohyby kurzu jednotlivých akcií. Postup prognózy se děje ve třech krocích [1]:

- 1) Globální analýza - zkoumá zejména celkovou hospodářskou situaci odvětvové tendence;
- 2) Odvětvová analýza - zkoumá odvětvové tendence;
- 3) Analýza konkrétní akcie - zkoumá vnitřní hodnotu akcie.

Fundamentální analýza má četné příznivce i stoupence. Mezi obhájce této metody patří například nositel Nobelovy ceny profesor Loist, který uvádí: „*Fundamentální analýzy se snaží zjistit vliv podnikových a globálních faktorů na vývoj akciových kurzů a využít je pro ziskovou investorskou strategii. Vzdor problémům s oceňováním poskytuje fundamentální analýza hodnotné údaje o budoucím vývoji podniku. Fundamentální ukazatelé patří k samozřejmému instrumentariu analytiků cenných papírů v praxi. Nové výzkumy ukázaly, že vyhodnocování ročních uzávěrek nelze ničím nahradit*“ [1].

Naopak nelichotivě se o fundamentálních analyticích vyjádřili např. R.D. Edwards a J. Magge: „*Je nesmyslné chtít určovat vnitřní hodnotu akcií. Statistická data, která fundamentální analytici studují, jsou minulost. Burzovní trh se nezajímá o minulost, ale o přítomnost. Burzovní kurz je určován pouze nabídkou a poptávkou. Burzovní kurz nereflektuje*

rozdílné hodnocení vnitřních hodnot substančních analytiků. Kurz reflektuje naděje, obavy, domněnky a nálady tisíců potenciálních kupujících a prodávajících“ [1].

Globální analýza prognózuje vývoj akciového trhu jako celku. Jejím cílem je odhadnout budoucí makroekonomické ukazatele a jejich vliv na akciové kurzy. Nejdůležitější ukazatele, kterým se globální analýza věnuje, jsou [1] hrubý domácí produkt a míra jeho růstu, fiskální politika, monetární politika, vliv úrokových sazeb, inflace, mezinárodní pohyb kapitálu a ekonomické a politické šoky.

Odvětvová analýza prognózuje vývoj v jednotlivých odvětvích a analyzuje jejich rozdílné charakteristiky. U každého odvětví se zjišťuje zejména nákladová stránka výroby, míra zisku, exportní schopnosti, inovační a technologická očekávání a jejich vliv na akciové kurzy. Jednotlivá odvětví se rozlišují podle toho, jak reagují na změnu produktu a agregátní poptávky na cyklická, neutrální a acyklická [1].

Fundamentální analýza předpokládá, že každá akcie má svou vnitřní hodnotu, která vychází z historických dat z hospodaření společnosti a aktuální kurz akcie se pohybuje kolem této vnitřní hodnoty. Tato analýza se skládá ze dvou postupných kroků, nejprve za použití finančních analýz se snaží zjistit vnitřní hodnotu a prognózovat budoucí vývoj společnosti, a pak tyto údaje srovnává s aktuálním kurzem a snaží se zjišťovat, zda je akcie podhodnocena (impuls k nákupu) nebo nadhodnocena (impuls k prodeji), což udržuje kurz kolem vnitřní hodnoty. V praxi se dnes používá relativních ukazatelů, které slouží k porovnávání akcií mezi sebou či v dlouhodobějších časových obdobích. Nejcitovanějším je ukazatel P/E ratio, tj. poměr ceny akcie a zisku na akcii.

1.4.2 Technická analýza

Technická analýza se snaží předpovědět pohyby kurzu jednotlivých akcií pomocí analýzy minulého vývoje kurzů, objemu obchodů a minulých trendů [1]. Je reakcí na kritiku fundamentální analýzy, její neschopnosti určit vnitřní hodnotu a zachytit kurzotvorné změny. Nedůvěřuje pramenům, podle kterých tvoří fundamentalisté své předpovědi, a vytýká jim časovou náročnost.

Metod technické analýzy je mnoho, ale všechny se odvíjejí od pěti základních principů [1]:

- 1) Kurzy akcií jsou určovány pouze nabídkou a poptávkou;
- 2) Nabídka a poptávka je ovlivňována racionálními i iracionálními činiteli;
- 3) Kurzy akcií mají tendenci pohybovat se v trendech po významné časové úseky;

- 4) Změny v trendech jsou způsobovány posuny poptávky a nabídky a lze je včas identifikovat studiem historických cen a objemu obchodů;
- 5) Mnoho grafických formací v grafech a diagramech má tendenci se opakovat, což umožňuje prognózovat budoucí vývoj kurzů akcií.

Množství rozličných metod technické analýzy se dá shrnout do tří základních skupin [1]:

a) *Vyhodnocování trendů* - zakladatel této teorie je americký podnikatel Charles H. Dow, který tvrdil, že většina akcií se chová na trzích podobným způsobem, zatímco velmi málo akcií vykazuje jiné chování. Rozlišoval tři druhy trendů [1]:

- 1) Primární trend - rozsáhlé vzestupné, či sestupné trendy trvající několik let;
- 2) Sekundární trend - méně významné kolísání trvající několik měsíců;
- 3) Terciární trend - nevýznamné fluktuace trvající několik dní, maximálně týdnů.

Trend se skládá ze čtyř základních fází - klesající fáze (Bear Market - medvědí trh), postranní fáze akumulací (Side Ways), růstová fáze (Bull Market - býčí trh) a postranní fáze distribuční (Side Ways) [1].

b) *Zkoumání grafů pomocí linií a formací* - podle této metody lze vyvodit, jestli kurz akcie bude pokračovat v trendu (trojúhelníky a klíny), nebo zda jej přeruší (vlajky a prapory) či změnění (hlava a ramena).

c) *Zkoumání klouzavých průměrů a indikátorů* - pomocí této metody lze získat informace o situaci na trhu, rozhodovacích signálech pro nákup či prodej, apod. Tyto signály mají upozornit na situaci, kdy je trh nasycen a lze očekávat brzký pokles nebo naopak, kdy je trh nenasyčen a lze očekávat oživení na trhu spojené s růstem kurzů. Používané indikátory jsou např. momentum, index relativní síly, linie růstu a poklesu, míra změny atd.

Klouzavý průměr konstruuje ceny v minulosti na základě lineární kombinace, k čemuž využívá různé koeficienty. Jejich součet je vždy roven jedné. Klouzavé průměry jsou jedním z nejpoužívanějších metod technické analýzy. Vyhlažují prudké výkyvy cen a identifikují trend. Nevýhodou je jejich vycházení z minulosti, a tedy jisté zpoždění za aktuálním vývojem. Zde je uveden přehled používaných klouzavých průměrů [1].

Jednoduchý klouzavý průměr je aritmetický průměr posledních pozorování. Délka zvoleného období pro výpočet průměru je volena na základě délky investice, charakteru trhu či

empirických výzkumů. Čím je kratší perioda, tím více je vysíláno signálů, čímž dochází ke zvýšení míry výskytu falešných signálů. *Trojúhelníkový klouzavý průměr*, u něhož je váha jednotlivých historických dat různá. Největší váhu mají data uprostřed periody, směrem ke krajům váhy lineárně klesají. U *váženého klouzavého průměru* jsou data v periodě rozložena lineárně. Největší váhu mají současné hodnoty. Obvykle se nejstarší hodnotě přidělí váha 1, druhé hodnotě váha 2 až do současnosti. V případě *exponenciálního klouzavého průměru* jsou váhy rozděleny exponenciálně. Největší váhu mají aktuální hodnoty, nejmenší váhu mají nejstarší hodnoty. Bere v úvahu veškerá data z minulosti a není omezen délkou periody. *Variabilní klouzavý průměr* je speciálním případem exponenciálního. Přidává k němu dvě vyrovnávací konstanty podle volatility ceny. Jeho výhodou je, že v období vysoké volatility se stává citlivějším a dává signály ve větší míře a rychleji. Naopak v okamžicích stability cen se stává klouzavý průměr méně citlivý. *Klouzavá regrese* se konstruuje odlišně oproti klouzavým průměrům. Využívá klouzavého prokládání přímky posledními n pozorováními metodou nejmenších čtverců.

Pásmová analýza je založena na klouzavých průměrech. Počítá tři křivky - klouzavý průměr ceny akcie, horní a dolní hranici pásma. Šířky pásma od klouzavého průměru jsou konstantní či proměnlivé v závislosti na volatilitě ceny. Rozlišují se tři základní pásma [1]:

- 1) Procentní;
- 2) Bolligerovo (BB) - šířka pásma se mění dle volatility akcie;
- 3) Pásky klouzavých průměrů (MAB) - šířka pásma se mění dle volatility klouzavého průměru.

Oscilátory jsou skupinou indikátorů, které měří změnu ceny za zvolený časový úsek. Sledují trendy, extrémy a divergence od cen. Používají se zejména při postranním pohybu cen, při trendech často vydávají falešné signály. Oscilátorů je používáno značné množství, např. [1]:

- 1) Momentum - porovnává dnešní ceny s cenou před n dny;
- 2) Rate of Change (ROC) - momentum vyjádřené v procentech;
- 3) Moving Average Spread (MAS) - rozpětí dvou klouzavých průměrů stejného typu a různé délky;
- 4) Cenový oscilátor - porovnává dnešní hodnoty exponenciálního průměru s hodnotou před n dny;
- 5) Index relativní síly (RSI);
- 6) Stochastické oscilátory;
- 7) Moving Average Convergence Divergence (MACD).

Technická analýza rozhodně není neomylný nástroj pro prognózování budoucích akciových kurzů a není prokázáno, že investoři investující podle technické analýzy dosahují dlouhodobě nadprůměrných výnosů. Pokud různé zpětné výpočty ukázaly, že za určitých podmínek a metod bylo možno v historii dosáhnout lepších výsledků, tak by tyto zlepšené výsledky stejně ani nestačily na zaplacení zvýšených transakčních poplatků. Nejsou žádné racionální důvody pro to, aby se opakovala historie podle sebesložitějších formací či výpočtů [1].

1.4.3 Psychologická analýza

Psychologická analýza je založena na předpokladu, že na kurz akcií mají velký vliv psychologické reakce investorů zejména v krátkodobém časovém horizontu. Zakládá si na předpokladu, že v ekonomii se nedá vše spočítat a reakce lidí nejsou založeny na racionálních úvahách. Proto se snaží odhalovat změny v psychologickém chování investorů. Nejvýznamnější představiteli jsou J. M. Keynes a A. Kostolany [1].

Teorie efektivních trhů (*Efficient market*) je v současnosti nejvýznamnějším proudem v oblasti financí. Teorie tvrdí, že akciový trh je efektivní a kurzotvorné informace v sobě obsahuje téměř okamžitě (rozšíření důležité informace trvá maximálně 30 vteřin). Proto selhávají jednotlivé obchodní strategie, protože kurz je vždy objektivní a přizpůsobuje se své vnitřní hodnotě a na pohyby kurzu má vliv pouze neočekávaná informace a proto se kurzy pohybují též neočekávaně. Předpoklady efektivního trhu jsou [1]:

- Na akciovém trhu působí obrovské množství racionálních investorů, kteří průběžně analyzují i obchodují.
- Investoři mají k dispozici dostatek pravdivých, levných a aktuálních informací.
- Investoři reagují přesně a rychle na nové informace.
- Transakční náklady jsou nízké.
- Trh je vysoce likvidní.
- Žádný účastník nemá monopolní či výsadní postavení.
- Kvalitní infrastruktura a právní regulace trhu.

Dle síly efektivnosti se rozlišují tři základní formy [1]:

- 1) Slabá forma efektivnosti - kurz obsahuje všechny historické informace, a proto prognózování kurzu na základě minulého průběhu kurzu je nemožné, čímž se zpochybňuje technická analýza.
- 2) Středně silná forma efektivnosti - kurz obsahuje všechny historické a současné veřejné informace, čímž se zpochybňuje i fundamentální analýza.

3) Silná forma efektivnosti - kurz obsahuje všechny informace včetně neveřejných.

Bylo provedeno mnoho výzkumů na téma teorie efektivních trhů, ale byla potvrzena pouze slabá a částečně středně silná forma efektivnosti. Je potvrzeno, že existují neveřejné informace přístupné pouze pro vyvolené profesionály, na jejichž základě dosahují nadprůměrných výnosů.

I přestože teorie efektivních trhů předpokládá, že kurz je vždy objektivní, dochází k pravidelně opakujícím se anomáliím, které však lze víceméně spolehlivě vysvětlit [1]:

- Akvizice a fúze - dochází k abnormálním výkyvům kurzů obvykle směrem vzhůru. Lze očekávat snížení nákladů podniku a z pohledu investorů snížení rizika.
- Kótování na burzách je doprovázeno růstem kurzů. Souvisí se zvýšenou likviditou na trhu a je vnímáno jako zvýšení kvality společnosti a tedy jako snížení rizika.
- Efekt malých společností - velikost společností je v inverzním vztahu s výnosem. Lze předpokládat, že investoři menší společnosti vyhledávají méně z důvodů nízké likvidity a vyššího rizika a proto jsou jejich ceny nižší než u větších společností ve stejné ekonomické situaci.
- Efekt nízkého P/E (poměr kurzu akcie a zisku po zdanění na jednu akcii) - čím nižší je P/E, tím vyšší je výnos. I toto je relativně logické, protože nízké P/E znamená, že na akcii připadá vyšší zisk, který je možno reinvestovat, či vyplatit více na dividendách a to se zákonitě pozitivně odrazí v budoucnosti ve formě vyššího výnosu.
- Pondělní efekt - v pondělí je výnos akciového trhu záporný. Pravděpodobně způsoben mimo jiné oznamováním negativních zpráv v pátek po uzavření obchodování, povíkendové nervozitě na trzích a přehodnocením svých investorských pozic přes víkend.
- Lednový efekt - v lednu je výnos na akciových trzích mimořádně vysoký. Za příčinu lednového efektu lze snad označit daňovou optimalizaci na konci roku, poklesy trhů v prosinci v důsledku vyšších prodejů akcií kvůli potřebě zvýšení disponibilních prostředků domácností před vánočními a podniků před roční uzávěrkou a vyplácením vánočních prémie.

1.5 Dílčí závěr

V praxi se používá kombinace uvedených akciových analýz. Investorská politika se určuje podle globální fundamentální analýzy, výběr titulů se uskutečňuje na základě mikroekonomické fundamentální analýzy a technická analýza slouží ke správnému načasování obchodu (*Timing*). I při pochybnostech o úspěšnosti těchto analýz se vyplatí je sledovat a odhadnout tak chování investorů, kteří se podle analýz řídí. V této práci budou využity popsané indikátory technické analýzy pro predikci ceny vybraného akciového indexu.

2 Posilované učení

Posilované učení je přístup k počítačové inteligenci, který kombinuje dvě disciplíny k úspěšnému řešení problémů, které žádná jiná disciplína nemůže vyřešit individuálně. Dynamické programování je oblast matematiky, která se tradičně používá k řešení problémů optimalizace a řízení. Nicméně, tradiční dynamické programování je omezené velikostí a složitostí problémů, které může řešit. Učení s učitelem je obecná metoda pro aproximaci parametrické funkce, jako jsou neuronové sítě. Nicméně, učení s učitelem vyžaduje vzorek vstupních a výstupních dvojic z funkce, které se má naučit. Posilované učení kombinuje oblasti dynamického programování a učení s učitelem. V posilovaném učení má počítač jednoduše daný cíl, který má dosáhnout. Počítač se pak učí jak dosáhnout tohoto cíle pokusem a omylem interakcemi s prostředím.

2.1 Historie posilovaného učení

Historie posilovaného učení má dva hlavní, bohaté prameny, které byly sledovány samostatně před propojením v moderním posilovaném učení. Jeden z nich se týká učení metodou pokusu a omylu a započal v psychologii zvířecího učení. Tento pramen vychází z některých prvních prací v oblasti umělé inteligence a vedl k oživení posilovaného učení na počátku 80. let. Druhý pramen se týká problému optimálního řízení a jeho řešení pomocí hodnotových funkcí a dynamického programování. Na konci 80. let došlo ke spojení uvedených přístupů a vytvoření moderní oblasti posilovaného učení [24].

Pojem "optimální řízení" se začal používat v 50. letech k popisu chování regulátorů dynamických systémů. Jeden z přístupů byl vyvinut v polovině 50. let Richardem Bellmanem a ostatními prostřednictvím rozšíření Hamiltonovi a Jacobiho teorie z devatenáctého století. Tento přístup využívá pojetí dynamického stavu systému a funkce hodnoty nebo optimální funkce výnosu k definování funkční rovnice, nyní často nazývané jako Bellmanova rovnice. Tato rovnice vyjadřuje, že užitek stavu přímo závisí na užitku okolních stavů (pokud agent volí optimální akci) a je dána vztahem [24]:

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s, a) U(s'), \quad (1)$$

kde $U(s)$ je užitek stavu vyjadřující dlouhodobé ocenění, $R(s)$ je krátkodobé ocenění, γ je diskontní faktor (faktor slevy) vyjadřující preference mezi aktuálním a budoucím oceněním

(blízko 0 = budoucnost není důležitá, blízko 1 = budoucnost je stejně důležitá jako současnost – aditivní funkce užitku), $P(s'|s,a)$ je pravděpodobnostní distribuce vyjadřující pravděpodobnost přechodu z s do s' akcí a a $U(s')$ je užitek ve stavu s' .

Skupiny metod pro řešení problémů optimálního řízení vyřešením této rovnice jsou známy jako dynamického programování. Bellman také představil diskrétní stochastickou verzi problému optimálního řízení, známého jako Markovské rozhodovací procesy (MDP) a Ron Howard vymyslel metodu iterace strategie pro MDP. Všechny z nich tvoří základní prvky, z nichž vychází teorie a algoritmy moderního posilovaného učení [24].

Dynamické programování je široce považováno za jediný možný způsob řešení všeobecných stochastických problémů optimálního řízení. Je postiženo tím, co Bellman nazval "prokletí rozměrnosti", což znamená, že jeho výpočetní náročnost roste exponenciálně s počtem stavových veličin. Dynamické programování bylo usilovně vyvíjeno od pozdních 50. let, včetně rozšíření částečně zjistitelných MDP podle průzkumu Lovejoye, mnoha aplikací podle průzkumu Whita, aproximačních metod podle průzkumu Rusta, a asynchronních metod od Bertseka [24].

Metoda učení pokusem a omylem se objevila v psychologii, kde jsou posilované teorie učení časté. Pravděpodobně první, kdo výstižně vyjádřil podstatu učení pokusem a omylem, byl Edward Thorndike. Vyjadřuje, že akce následovaná dobrými nebo špatnými výsledky má tendenci být znovu zvolena a odpovídajícím způsobem změněna. Thorndike toto nazval Law of effect¹, protože popisuje vliv posílení událostí v tendenci k výběru akcí [24].

V roce 1989 byly oblasti Temporal Difference (TD) a optimálního řízení plně spojeny s rozvojem Q-učení Chrisem Watkinsem. Tato práce rozšířila a sjednotila předchozí výzkum všech tří oblastí výzkumu posilovaného učení (tj. včetně TD). Až do uveřejnění Watkinsovi práce byl ohromný nárůst ve výzkumu posilovaného učení, především strojového učení jako podoblasti umělé inteligence, ale také ve výzkumu neuronových sítí a umělé inteligence v širším měřítku. V roce 1992 pak pozoruhodný úspěch vrhcábů (backgammon)² Gerry Tesaura jako hracího programu přitáhl další pozornost k posilovanému učení [24].

¹ Účinek jednání ve skutečnosti rozhoduje, zda se stane reakce na daný podnět nebo ne. Pozitivní účinky chování vedoucí k uspokojení z úspěchu posilují spoj mezi podnětem a reakcí, negativní důsledky pak daný spoj zeslabují.

² Backgammon je jedna z nejstarších deskových her pro dva hráče. Herní deska se skládá ze čtyř kvadrantů, které jsou rozděleny do 6 oblastí. Cílem hry je, aby se vaše dáma dostala domů co nejrychleji. Štěstí hraje důležitou roli ve hře, ale je velmi obtížné vyhrát bez strategie.

2.2 Markovský rozhodovací proces

Úloha posilovaného učení, která splňuje požadavky Markovské vlastnosti, se nazývá Markovský rozhodovací proces (MDP). Jestliže jsou prostory stavů a akcí konečné, potom je nazývána konečný MDP. Konečné MDP jsou zvláště důležité pro teorii posilovaného učení [26].

MDP [26] je pojmenován po ruském matematikovi Andreji Markovovi. Poskytuje matematický rámec pro modelování rozhodování v situacích, kdy jsou výsledky zčásti náhodné a zčásti pod kontrolou uživatele. MDP se využívají pro studium mnoha typů optimalizačních problémů řešených prostřednictvím dynamického programování a posilovaného učení. MDP jsou známy od 50. let 20. století. Dnes jsou využívány v různých oblastech včetně robotiky, automatizovaného řízení, ekonomie a průmyslové výroby.

Přesněji řečeno MDP je diskrétní, stochastický a kontrolovaný proces. V každém časovém okamžiku je proces v určitém stavu s a uživatel může vybrat jakoukoli akci a , která je dostupná ve stavu s . Proces na tuto akci v následujícím časovém okamžiku reaguje náhodným přesunutím do nového stavu s' a dává uživateli odpovídající užitek $R_a(s,s')$ [26].

Pravděpodobnost, že proces vybere s' jako nový stav, je ovlivněna vybranou akcí. Pravděpodobnost je určena funkcí přechodu stavu $P_a(s,s')$. Takže následující stav s' závisí na současném stavu s a na uživatelově akci a . Dané s a a jsou však podmíněně závislé na všech předchozích stavech a akcích. Jinými slovy má přechod stavu MDP Markovskou vlastnost.

MDP jsou rozšířením Markovských řetězců³; rozdíl je v přidání akcí (umožňují výběr) a užitků (motivace). Pokud by existovala pouze jedna akce, nebo pokud by byla daná uskutečnitelná akce stejná pro všechny stavy, MDP by se zredukoval na Markovský řetězec [26]. MDP je uspořádaná čtveřice $(S, A, P(.,.), R(.,.))$, kde [26]:

- S je konečná množina stavů,
- A je konečná množina akcí (alternativa: A_s je konečná množina akcí dostupných ve stavu s),
- $P_a(s,s')$ je pravděpodobnost přechodu, že akce a ve stavu s v čase t povede do stavu s' v čase $t + 1$,

³ Markovský řetězec je náhodný proces s diskrétní množinou stavu, diskrétním časem a takový, že pravděpodobnost $p(i)$, že v čase t bude proces ve stavu i , je stochasticky závislá pouze na stavu v předchozím okamžiku, tj. na stavu v čase $t - 1$.

$$P_a(s, s') = P_r\{s_{t+1} = s' \mid s_t = s, a_t = a\} \quad (2)$$

- $R_a(s, s')$ je okamžitý užitek (ne očekávaný okamžitý užitek) dosažený po přechodu stavu na s' ze stavu s s pravděpodobností přechodu $P_a(s, s')$.

Hlavním úkolem MDP je najít strategii pro uživatele, funkci π , která specifikuje akci $\pi(s)$ zvolenou uživatelem ve stavu s . Je třeba poznamenat, že je-li jednou MDP takto zkombinován se strategií, je pro každý stav určena akce a výsledná kombinace se chová jako Markovský řetězec [26]. Cílem je vybrat strategii π , která bude maximalizovat kumulativní funkci náhodných užiteků, typicky očekávanou diskontovanou sumu za potenciálně nekonečnou dobu [26]:

$$\sum_{t=0}^{\infty} \gamma^t R_{a_t}(s_t, s_{t+1}) \quad (\text{kde } a_t = \pi(s_t)) \quad (3)$$

kde γ je diskontní faktor splňující $0 < \gamma \leq 1$ a typicky se blíží 1. Díky Markovské vlastnosti může být optimální strategie pro tento speciální problém skutečně zapsána jako funkce pouze s .

Nechť je známa funkce přechodu stavu P a užitková funkce R . Cílem je zjistit strategii, která maximalizuje očekávanou diskontovanou hodnotu. Standardní skupina algoritmů používaná pro zjištění této optimální strategie vyžaduje uložení dvou polí indexovaných stavem: hodnota V , která obsahuje reálné hodnoty, a strategie π , která uchovává akce. Na konci algoritmu bude π obsahovat řešení a $V(s)$ bude obsahovat diskontovanou sumu užiteků, které budou (v průměru) získány provedením tohoto řešení ve stavu s [26].

Algoritmus má následující dva druhy kroků, které jsou opakovány v určitém pořadí pro všechny stavy, dokud nedojde k nějakým změnám [26].

$$\pi(s) := \arg \max_a \{R_a(s, s') + \gamma \sum_{s'} P_a(s, s') V(s')\}, \quad (4)$$

$$V(s) := R(s) + \gamma \sum_{s'} P_{\pi(s)}(s, s') V(s'), \quad (5)$$

kde $R(s)$ je zde $R_{\pi(s)}(s)$. Pořadí kroků závisí na variantě algoritmu. Kroky se buď mohou provádět pro všechny stavy najednou, nebo pro každý stav popořadě, a pro některé stavy častěji než pro jiné. Dokud není nějaký stav permanentně vyloučen z nějakého z kroků, algoritmus má šanci dospět ke správnému řešení.

2.3 Základní pojmy posilovaného učení

Kromě agenta a prostředí lze rozlišit několik základních prvků systému posilovaného učení jako je strategie, funkce odměny, hodnotová funkce a případně i model prostředí.

Strategie [24] definuje způsob učení agenta, jak se chovat v daném čase. Zhruba řečeno, strategie je mapování z vnímaných stavů prostředí k akcím, které mají být přijaty v těchto stavech. To odpovídá tomu, co v psychologii bývá nazýváno soubor pravidel podnětů a reakcí nebo sdružení. V některých případech strategie může být jednoduchá funkce nebo vyhledávací tabulka, zatímco v jiných může zahrnovat rozsáhlý výpočet, jako proces hledání. Strategie je základem pro agenta posilovaného učení ve smyslu, že sama o sobě stačí k určení chování. Obecně platí, že strategie mohou být stochastické.

Funkce odměny [24] definuje cíl v problému posilovaného učení. Zhruba řečeno, mapuje každý vnímaný stav (nebo dvojici stav a akce) prostředí jako jedno číslo, odměnu, označující vnitřní vhodnost tohoto stavu. Jediným cílem agenta posilovaného učení je maximalizovat celkovou odměnu, kterou obdrží v dlouhodobém měřítku. Funkce odměny definuje, jaké jsou dobré a špatné události pro agenta.

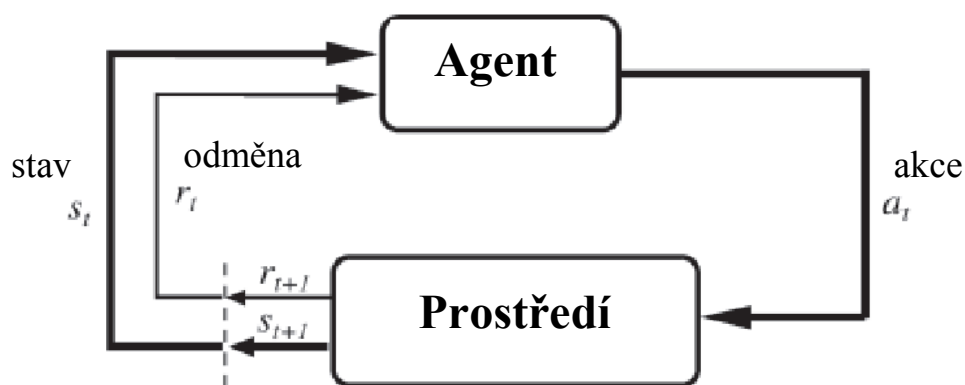
Zatímco funkce odměny označuje co je dobré v okamžitém smyslu, *hodnotová funkce* [24] určuje co je dobré v dlouhodobém měřítku. Zjednodušeně řečeno, hodnota stavu, je celková výše odměny agenta, kterou může očekávat, že nahromadí v budoucnosti, počínaje tímto stavem. Vzhledem k tomu, že odměny určují okamžitou, vnitřní vhodnost stavů prostředí, hodnoty označují dlouhodobou vhodnost stavů po zohlednění stavů, které mohou následovat a odměny dostupné v těchto stavech. Například stav by mohl vždy vynášet nízkou okamžitou odměnu, ale stále mít vysokou hodnotu, protože je pravidelně následován dalšími stavy, které přinášejí vysoké odměny. Nebo opak může být pravdou. V lidské analogii mohou být odměny chápány jako potěšení (pokud jsou vysoké) a bolest (pokud jsou nízké). Zatímco hodnoty odpovídají přesnějším a prozíravějším úsudku o tom, jak je agent potěšen nebo nespokojen s tím, že jeho prostředí je v určitém stavu.

2.3.1 Rozhraní agent - prostředí

Problematika posilovaného učení je zamýšlena jako přímočaré vymezení problému učení se ze vzájemného ovlivňování za účelem dosažení cíle. Učící se a tvůrce rozhodnutí je nazýván *agent*. Věc, se kterou se vzájemně ovlivňuje, zahrnuje vše mimo agenta a je nazývána *prostředím*. Tyto interakce probíhají neustále. Agent vybírá akce a prostředí reaguje

na tyto akce tím, že předkládá agentovi nové situace. Prostředí také vyvolává odměny, zvláštní číselné hodnoty, které se agent snaží maximalizovat v průběhu času [24].

Přesněji řečeno, interakce mezi agentem a prostředím probíhají v každé posloupnosti diskrétních časových kroků $t=0,1,2,3,4 \dots$. V každém časovém kroku t agent obdrží nějaké vyobrazení stavu prostředí $S_t \in S$, kde S je soubor možných stavů, a na tomto základě vybírá akce $a_t \in A(S_t)$, kde $A(S_t)$ je soubor akcí, které jsou k dispozici ve stavu S_t . O jeden krok později, částečně jako důsledek své činnosti, agent obdrží numerickou odměnu $r_{t+1} \in R$ a ocitá se v novém stavu S_{t+1} [24]. Následující obrázek zobrazuje agenta v interakci s okolím.



Obrázek 1 - Interakce agenta s prostředím. Zdroj: [24]

V každém kroku, agent provádí mapování ze stavů k pravděpodobnostem výběru každé možné akce. Toto mapování je nazýváno strategie agenta a je označována π_t , kde $\pi_t(s, a)$ je pravděpodobnost, že $a_t = a$ pokud $s_t = s$. Metody posilovaného učení stanovují, jak agent mění svou strategii v důsledku svých zkušeností. Agentův cíl je maximalizovat celkové množství odměny, které obdrží v dlouhodobém horizontu [24].

Tato struktura je abstraktní a flexibilní a může být aplikována na mnoho různých problémech a mnoha různými způsoby. Například časové kroky nemusí odpovídat pevně stanoveným intervalům v reálném čase, mohou se vztahovat k libovolným po sobě následujícím etapám tvorby rozhodnutí a jednání. Akce mohou být nízké úrovně řízení nebo na vysoké úrovni řízení. Stejně tak stavy mohou přijmout celou řadu forem. Mohou být kompletně určeny vjemy nízké úrovně nebo mohou být více na vysoké úrovni a abstraktní. Některé mohou vycházet ze vzpomínek vjemů z minulosti, nebo dokonce být výhradně duševní nebo subjektivní. Podobně, některé akce mohou být zcela duševní nebo výpočtové. Obecně lze říct, že akce mohou být jakákoli rozhodnutí, které se chce agent naučit provést, a stavy mohou být cokoli, co ví, že by mohlo být užitečné při jejich tvorbě [24].

Obecné pravidlo říká, že co nelze změnit libovolně agentem, není považováno za jeho součást, a je tedy součástí jeho prostředí. Nepředpokládá se, že všechno v prostředí je agentovi neznámé. Například agent často ví docela dost o tom, jak jsou jeho odměny počítány jako funkce jeho akcí a stavů. Ale vždy je výpočet odměny považován za externí, protože definuje úkol, kterému agent čelí, a proto musí být mimo jeho schopnost libovolně ho měnit. Ve skutečnosti v některých případech agent může vědět všechno o tom, jak pracuje jeho prostředí a stále čelí obtížné úloze posilovaného učení. Hranice mezi agentem a prostředím představuje omezení agentovy absolutní kontroly nikoli jeho znalostí. Tato hranice může být umístěna na různých místech pro různé účely. V praxi je hranice určena jakmile vybere konkrétní stavy, akce a odměny, a tím určí specifickou úlohu tvorby rozhodnutí, která je předmětem zájmu [24].

System posilované učení představuje významnou abstrakci problému cílem řízeného učení se ze vzájemného ovlivňování. Jakýkoli problém učení cílem řízeného chování může být redukován na tři signály procházejícími tam a zpět mezi agentem a jeho prostředím. Jeden signál reprezentuje volby provedené agentem (akce), další signál představují základ, na kterém jsou volby provedené (stavy) a třetí signál definuje agentův cíl (odměny). Tento rámec nemusí být dostatečně účinný k reprezentaci všech rozhodovacích problémů učení, ale ukázalo se, že může být velmi užitečný a použitelný.

2.3.2 Cíle a odměny

V posilovaném učení, je účel nebo cíl agenta formalizován z hlediska zvláštního signálu odměny, který přichází k agentovi z prostředí. V každém kroku je odměna jednoduché číslo $r_t \in R$. Neformálně je cílem agenta maximalizovat celkovou výši odměny, kterou obdrží. Znamená to maximalizovat nikoli okamžitou odměnu, ale tu kumulativní odměnu v dlouhodobém výhledu [24].

Využití odměny k formalizaci myšlenky cíle je jedním z nejvýraznějších rysů posilovaného učení. Přestože tento způsob stanovení cílů se může zdát na první pohled omezující, v praxi se ukázal být flexibilní a široce použitelný. Nejlepším způsobem, jak na to nahlížet je zvážit příklady jak by toho mohlo být využito. Například přimět robota naučit se chodit. Výzkumníci poskytovali odměnu v každém kroku úměrně k pokroku robota vpřed. Při učení robota jak uniknout z bludiště, dokud neunikne je odměna často nulová, až když se mu to podaří tak +1. Další společný postup v tomto učení je poskytnout odměnu -1 v každém časovém kroku, učiněném před únikem, což vybízí agenta k co nejrychlejšímu úniku. Člověk by také mohl chtít dát robotu negativní odměnou, např. když narazí do věcí. Pro agenta učícího se hrát

dámu nebo šachy přirozenou odměnou +1 je vítězství, -1 za prohru a 0 pro remízu a pro všechny nekoncové pozice [24].

Agent se vždy učí maximalizovat svůj signál odměny. Pokud má být dosaženo cíle, musí být odměny poskytnuty takovým způsobem, aby jejich maximalizací agent dosáhl také hlavních cílů. Je tedy důležité, aby stanovené odměny skutečně signalizovaly cíl. Zejména signál odměny není místem vtisknutí agentovi předchozí znalosti o tom, jak dosáhnout hlavního cíle. Například šachy hrající agent by měl být odměněn jen za skutečné vítězství, a ne za dosažení dílčích cílů, jako jsou soupeřovy figurky či ovládnutí středu desky. Pokud by byl za dosažení těchto dílčích cílů odměněn, potom by mohl najít způsob, jak jich dosáhnout, aniž by dosáhl skutečného cíle [24].

2.3.3 Výnosy

Pokud jde o cíl učení, až dosud nebyl přesně definován. Jak bylo uvedeno, agentův cíl je maximalizovat odměnu, kterou obdrží z dlouhodobého hlediska. Pokud posloupnost odměny obdržená po časovém kroku t je označována $r_{t+1}, r_{t+2}, r_{t+3}, \dots, r_T$, potom otázkou je, jak upřesnit aspekt této posloupnosti, který má být maximalizován? Obecně platí, že je snahou maximalizovat očekávaný výnos R_t definovaný jako některé specifické funkce posloupnosti odměny. V nejjednodušším případě je výnos součet odměny [24]:

$$R_t = r_{t+1} + r_{t+2} + r_{t+3} + \dots + r_T, \quad (6)$$

kde T je konečný časový krok. Tento přístup má smysl v aplikacích, ve kterých je přirozená představa o konečném kroku. To znamená, že se interakce agenta s okolím přirozeně rozloží na pod posloupnosti, které se nazývají epizody (jako jsou jednotlivé mezihry hry, cesty bludištěm atd.). Každá epizoda končí ve zvláštním stavu nazvaném koncový stav, po kterém následuje obnovení do standardního počátečního stavu nebo do vzorku ze standardního rozdělení počátečních stavů. Úkoly s epizodami tohoto typu se nazývají epizodické úlohy. V epizodické úloze je někdy potřeba rozlišit soubor všech nekoncových stavů S z množiny všech stavů plus koncový stav, označovaný S^+ [24].

Na druhou stranu se v mnoha případech interakce agenta s okolím nerozloží přirozeně do identifikovatelných epizod, ale pokračuje nepřetržitě bez omezení. Nazýváme je tak pokračující úlohy. Formulace výnosu je problematická pro pokračující úlohy, protože konečný časový krok by měl být $T = \infty$ a výnos by mohl být snadno sám o sobě nekonečný.

Další pojem, který je třeba zavést, je *diskontování*. Podle tohoto přístupu, se agent pokusí vybrat akce tak, aby součet diskontovaných odměn, které obdrží v budoucím průběhu, byl

maximalizován. Zejména si zvolí akce a_t k maximalizaci očekávaného diskontovaného výnosu [24]:

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}, \quad (7)$$

kde γ je parametr, $0 \leq \gamma \leq 1$, nazvaný diskontní faktor.

Diskontní faktor určuje současnou hodnotu budoucí odměny: odměna obdržena po k časových krocích v budoucnu stojí jen γ^{k-1} násobek hodnoty, kterou by měla, pokud by byla obdržena okamžitě. Pokud $\gamma < 1$, nekonečný součet má konečnou hodnotu po dobu, kdy bude posloupnost odměny $\{r_k\}$ omezená. Pokud $\gamma = 0$, agent je krátkozraký v tom, že se snaží maximalizovat okamžité odměny. Jeho cíl v tomto případě je naučit se, jak si vybrat a_t , aby se maximalizovala jen r_{t+1} . Jestliže každá z agentových akcí co nastane, má vliv pouze na bezprostřední odměnu, a ne budoucí odměny, potom by je mohl krátkozraký agent maximalizovat odděleně maximalizováním každé okamžité odměny. Ale obecně maximalizace okamžité odměny může snížit přístup k budoucí odměně, takže výnos může být ve skutečnosti snížen. Jak se γ blíží k 1, cíl bere budoucí odměny v úvahu silněji a agent se stává více prozíravý [24].

2.3.4 Hodnotové funkce

Téměř všechny algoritmy posilovaného učení jsou založeny na odhadu hodnotových funkcí - funkcí stavů (nebo dvojic stav akce), které odhadují, jak je dobré pro agenta být v daném stavu (nebo jak je dobré provést určitou akci v daném stavu). Pojem "jak dobré" je definován z hlediska budoucích odměn, které lze očekávat nebo pro upřesnění, z hlediska očekávaného výnosu. Odměny, které agent může očekávat, že obdrží, v budoucnu závisí na přijatých akcích. V důsledku tohoto jsou hodnotové funkce definovány s ohledem na konkrétní strategie. Neformálně je hodnota stavu s v rámci strategie π označena jako očekávaný výnos $V^\pi(s)$ při zahájení s a poté následující π . Pro MDP lze definovat $V^\pi(s)$ formálně jako [24]:

$$V^\pi(s) = E_\pi \{R_t \mid s_t = s\} = E_\pi \left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s \right\}, \quad (8)$$

kde $E_\pi \{\}$ označuje očekávanou hodnotu za předpokladu, že agent následuje strategii π , a t je libovolný časový krok. Hodnota koncového stavu, pokud existuje, je vždy nulová. V^π se nazývá funkce stavové hodnoty pro strategii π [24]. Obdobně se definuje hodnota přijaté akce a ve stavu s v rámci strategie π , označenou $Q^\pi(s,a)$, jako očekávaný výnos začínajícím v s , přijetím akce a , a poté následováním strategie π [24]:

$$Q^\pi(s,a) = E_\pi \{R_t \mid s_t = s, a_t = a\} = E_\pi \left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a \right\}, \quad (9)$$

kde Q^π je funkce akční hodnoty pro strategii π . Hodnotové funkce V^π a Q^π lze odhadnout ze zkušenosti. Například, pokud agent následuje strategii π a udržuje průměr aktuálních výnosů (pro každý dosažený stav), které následovaly po tomto stavu, pak průměr bude konvergovat ke stavové hodnotě $V^\pi(s)$. Pokud samostatné průměry jsou uchovávány pro každou akci přijatou ve stavu, pak tyto průměry budou podobně konvergovat k akčním hodnotám $Q^\pi(s,a)$. Nazývají se metody odhadu typu Monte Carlo metod, protože zahrnují průměrování přes mnoho náhodných vzorků skutečných výnosů. Tyto druhy metod jsou uvedeny v další kapitole. Pokud existuje velmi mnoho stavů, pak nemusí být praktické zachovávat oddělené průměry jednotlivě pro každý stav. Místo toho by musel agent udržovat V^π a Q^π jako parametrické funkce a nastavit parametry, aby lépe odpovídaly pozorovaným výnosům. Toto může také produkovat přesné odhady, ačkoli hodně záleží na povaze aproximátoru parametrické funkce [24].

2.3.5 Optimální hodnotové funkce

Řešením úloh posilovaného učení se rozumí nalezení strategie, která dosahuje velkou odměnu v dlouhodobém horizontu. Pro konečné MDP, lze přesně definovat optimální strategii následujícím způsobem. Hodnotové funkce definují částečné uspořádání ve strategiích [24]. Strategie π je definována, že má být lepší než nebo rovna strategii π' , pokud jeho očekávaný výnos je větší nebo stejný s tou v π' pro všechny stavy. Jinými slovy $\pi \geq \pi'$, právě tehdy, když $V^\pi(s) \geq V^{\pi'}(s)$ pro všechny $s \in S$. Existuje vždy alespoň jedna strategii, která je lepší nebo je rovna všem ostatním strategiím. Toto je optimální strategie. Ačkoli může existovat více než jedna, všechny optimální strategie se označují π^* . Sdílejí stejnou funkci stavové hodnoty, tzv. optimální funkci stavové hodnoty, označenou V^* a definovanou [24]:

$$V^*(s) = \max_{\pi} V^\pi(s), \quad (10)$$

pro všechny $s \in S$. Optimální strategie také sdílejí stejnou optimální funkci akční hodnoty, označenou Q^* , a definovanou jako [24]:

$$Q^*(s,a) = \max_{\pi} Q^\pi(s,a), \quad (11)$$

pro všechny $s \in S$ a $a \in A(s)$. Pro dvojici stav a akce (s, a) dává tato funkce očekávaný výnos pro přijetí akce a ve stavu s , a poté následující optimální strategii. Lze tedy zapsat Q^* pomocí V^* následovně [24]:

$$Q^*(s,a) = E \{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\} \quad (12)$$

2.4 Elementární metody řešení

V této části práce jsou popsány základní skupiny metod pro řešení problému posilovaného učení: dynamické programování, Monte Carlo metody, Temporal Difference učení a genetické posilované učení.

2.4.1 Dynamické programování

Termín dynamické programování (DP) odkazuje na soubor algoritmů, které lze použít pro výpočet optimálních strategií daných dokonalým modelem prostředí označovaný jako Markovský proces rozhodování (MDP). Klasické DP algoritmy jsou omezeně použitelné v posilovaném učení z důvodu jejich předpokladu dokonalého modelu a vzhledem k jejich velké výpočetní náročnosti. Jsou ale stále důležité teoreticky. DP poskytuje nezbytný základ pro pochopení uvedených metod. Počínaje touto kapitolou se předpokládá, že prostředí je konečný MDP. To znamená, že jeho množiny stavů a akcí, S a A , pro $s \in S$ jsou konečné, a že jeho dynamika je dána souborem přechodových pravděpodobností [24]:

$$P_{ss'}^a = P_r\{s_{t+1} = s' \mid s_t = s, a_t = a\}, \quad (13)$$

a předpokládané okamžité odměny [24]:

$$R_{ss'}^a = E\{r_{t+1} \mid a_t = a, s_t = s, s_{t+1} = s'\}, \quad (14)$$

pro všechny $s \in S$, $a \in A(s)$, a $s' \in S^+$ (kde S^+ je S zvýšený o konečný stav, pokud je problém epizodický). Klíčovou myšlenkou DP a posilovaného učení obecně, je použití hodnotových funkcí k uspořádání a struktuře hledání dobrých strategií. Lze tak snadno získat optimální strategie, jakmile jsou nalezeny optimální hodnotové funkce V^* a Q^* , které splňují Bellmanovu rovnici optimality [24]:

$$\begin{aligned} V^*(s) &= \max_a E\{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\} \\ &= \max_a \sum_{s'} P_{ss'}^a [R_{ss'}^a + \gamma V^*(s')], \end{aligned} \quad (15)$$

$$Q^*(s, a) = E\{r_{t+1} + \gamma \max_{a'} Q^*(s_{t+1}, a') \mid s_t = s, a_t = a\}$$

$$= \sum_{s'} P_{ss'}^a [R_{ss'}^a + \gamma_{a'}^{max} Q^*(s', a')], \quad (16)$$

pro všechny $s \in \mathcal{S}$, $a \in A(s)$ a $s' \in \mathcal{S}^+$. DP algoritmy jsou získané otočením Bellmanových rovnic, jako jsou tyto do přiřazení, to znamená do aktualizace pravidel pro zlepšení aproximace požadovaných hodnot funkcí. Hodnocení strategie odkazuje na iterativní výpočet hodnotových funkcí pro danou strategii. Zlepšení strategie se odkazuje na výpočet zlepšené strategie dané hodnotou funkcí pro tuto strategii. Klasické metody DP pracují v cestách přes stavovou množinu, provádějící úplnou zálohovací operaci v každém stavu. Každá záloha aktualizuje hodnotu jednoho stavu na základě hodnot ze všech možných následných stavů a jejich pravděpodobnosti výskytu. Plné zálohy jsou úzce spojeny s Bellmanovou rovnicí tak, že jsou trochu více než tyto rovnice změněny pro přiřazení. Když zálohy již nevyústí v žádné změny hodnoty, konvergenčí jsou získané hodnoty, které splňují odpovídající Bellmanovu rovnici [24].

Stejně tak jako existují čtyři základní hodnotové funkce (V^π, V^*, Q^π a Q^*), existují čtyři Bellmanovy odpovídající rovnice a čtyři odpovídající plné zálohy. Intuitivní pohled na činnost záloh je dán zálohovacími diagramy.

DP metody, stejně jako téměř všechny metody posilovaného učení, mohou být chápány jako všeobecné iterace strategie GPI⁴ (Generalized Policy Iteration). GPI je obecná představa dvou interagujících procesů točících se kolem přibližné strategie a přibližné hodnotové funkce. Jeden proces bere strategii jako dané a vykonává nějakou formu hodnocení strategie, změnou hodnotové funkce, aby se více podobaly skutečné hodnotové funkci pro strategii. Další proces zastává hodnotou funkci jako danou a vykonává nějakou formu strategie zlepšení, změnou strategie ke zlepšení, za předpokladu, že hodnotová funkce je jeho hodnotou funkcí. Ačkoliv každý proces mění základ pro další, celkově spolu spolupracují pro nalezení společného řešení, strategie a hodnotové funkce, které jsou nezměněné žádným procesem, a proto jsou optimální [24]. Všechny DP metody aktualizují odhady hodnot stavů na základě odhadů hodnot nástupnických stavů. To znamená, že aktualizují odhady na základě jiných odhadů.

⁴ Při iteraci strategie je první krok proveden jednou a druhý krok je opakován, dokud nekonverguje. Poté je znovu jednou proveden první krok a tak stále dokola. Místo toho, aby byl druhý krok opakován, dokud nekonverguje, měl by být problém formulován a řešen jako množina lineárních rovnic. Tato varianta má výhodu, že zde existuje konečná podmínka ukončení: algoritmus je dokončen, pokud se pole π nezmění ve směru působení prvního kroku na všechny stavy.

Tato obecná podstata je nazývána jako bootstrapping. Mnoho metod posilovaného učení provádí bootstrapping, a to i těch, které je nevyžadují, jako DP metody vyžadují, úplný a přesný model prostředí. V další kapitole jsou představeny metody posilovaného učení, které nevyžadují model a ani neprovádějí bootstrapping. Tyto klíčové funkce a vlastnosti jsou oddělitelné, přesto mohou být smíšené v zajímavých kombinacích.

2.4.2 Monte Carlo metody

Monte Carlo metody vyžadují pouze zkušenost v podobě vzorku sekvence stavů, akcí a odměn z on-line nebo simulované interakce s prostředím. Učení se z on-line zkušeností je pozoruhodné, protože nevyžaduje předchozí znalost dynamik prostředí, přesto může stále ještě dosáhnout optimálního chování. Učení se ze simulované zkušenosti je také silné. I když je model vyžadován, potřebujete pouze generovat vzorek přechodů ne kompletní rozdělení pravděpodobnosti všech možných přechodů, který je vyžadován metodami dynamického programování. V mnoha případech je snadné vytvářet zkušenosti vzorku podle požadované pravděpodobnosti rozdělení, ale nemožné získat rozdělení v explicitní formě [24].

Monte Carlo metody jsou způsoby řešení problému posilovaného učení na základě průměrování vzorku výnosů. Pro zajištění, že jsou dobře definované výnosy dostupné, se definují metody Monte Carlo pouze pro epizodické úkoly. To znamená, že se předpokládá zkušenost rozdělená na epizody, a že všechny epizody posléze skončí bez ohledu na to, jaké akce jsou vybrány. Pouze při dokončení epizody jsou odhady hodnot a strategií změněny [24]. Navzdory rozdílům mezi Monte Carlo a DP metodami, z nichž nejdůležitější myšlenky se přenášejí od DP k případu Monte Carlo. Nejen, že počítají stejné hodnotové funkce, ale interagují k dosažení optimality v podstatě stejným způsobem. Stejně jako v kapitole DP, je uvažována první strategie hodnocení, výpočet V^π a Q^π za pevnou libovolnou strategií π , pak strategie zlepšení, a konečně všeobecná iterace strategie. Každá z těchto myšlenek převzatá z DP je rozšířena pro Monte Carlo případ, ve kterém je dostupný pouze vzorek zkušenosti [24]. Monte Carlo metody se učí hodnotové funkce a optimální strategie ze zkušenosti ve formě vzorkových epizod. To jim dává minimálně tři druhy výhod oproti DP metodám [24]. Za prvé, mohou být použity pro učení optimálního chování přímo z interakce s prostředím, bez modelu dynamik prostředí. Za druhé, mohou být použity k simulaci nebo vzorkovým modelům. Za třetí, je snadné a efektivní zaměřit se na Monte Carlo metody v malé podmnožině stavů. Oblast zvláštního zájmu je možné přesně ohodnotit, aniž by to bylo na úkor přesného hodnocení zbytku stavové množiny. Další výhodou metod Monte

Carlo je, že mohou být méně poškozené narušením Markovské vlastnosti. Je to proto, že neaktualizují svou hodnotu odhadů na základě hodnoty odhadů nástupnických stavů.

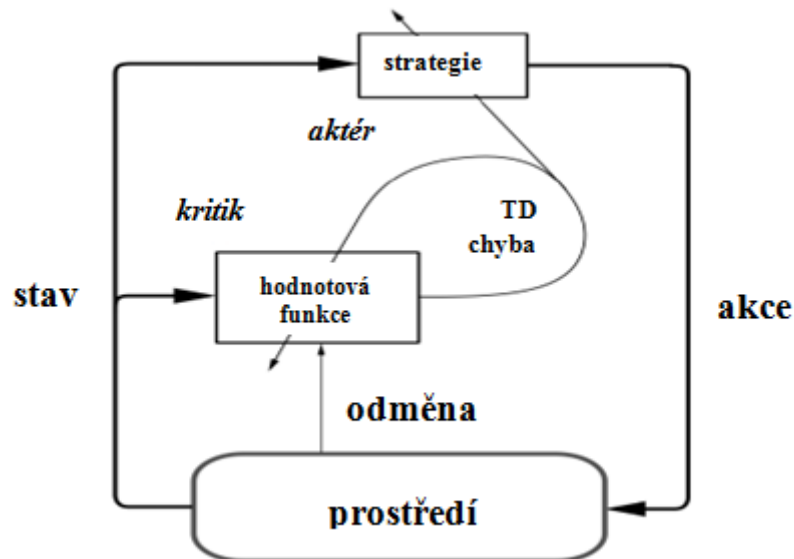
Při navrhování Monte Carlo metod, je sledováno celkové schéma všeobecné iterace strategie (GPI). GPI zahrnuje interagující procesy strategie hodnocení a strategie zlepšování. Monte Carlo metody poskytují alternativní strategii hodnocení procesu. Spíše než použití modelu pro výpočet hodnoty každého stavu, jednoduše průměrují mnoho výnosů začínajících ve stavu. Protože stavová hodnota je očekávaný výnos, může se tento průměr stát dobrou aproximací hodnoty. Monte Carlo metody kombinují strategii hodnocení a strategii zlepšení kroků na základě postupných epizod, a mohou být takto přírůstkově realizovány na základě postupných epizod [24].

2.4.3 Temporal Difference učení

Ústřední a novou myšlenkou v posilovaném učení je nepochybně Temporal Difference (TD) učení. TD učení je kombinací Monte Carlo myšlenek a dynamického programování. TD metody se mohou učit přímo ze surové zkušenosti bez modelu dynamiky prostředí stejně jako Monte Carlo metody. DP i TD metody aktualizují odhady založené z části na jiných naučených odhadech, bez čekání na konečný výsledek. Vztah mezi TD, DP a Monte Carlo metodami je stále se opakujícím tématem v teorii posilovaného učení. Tyto myšlenky a metody vzájemně se prolínající lze mezi sebou kombinovat mnoha způsoby [24]. TD metody se učí své odhady částečně na základě jiných odhadů. Učí se odhadovat z odhadů a jsou tedy metody provádějící bootstrapping. Je zřejmé, že TD metody mají výhodu oproti metodám DP v tom, že nevyžadují model prostředí, jeho odměny a rozdělení pravděpodobnosti následujícího stavu. Další zřetelnou výhodou metod TD před metodami Monte Carlo je skutečnost, že jsou přirozeně prováděny on-line, plně přírůstkovým způsobem. U Monte Carlo metod je nutné počkat až do konce epizody, protože jedině pak je známý výnos, zatímco u metod TD stačí čekat pouze jeden časový krok.

Pro každou fixní strategii algoritmu TD je prokázána konvergence k V^π , v průměru pro konstantní velikost kroku parametru, pokud je dostatečně malý a s pravděpodobností 1, pokud se velikost kroku parametru snižuje podle obvyklých stochastických aproximačních podmínek. TD metody jsou alternativním řešením problému predikce Monte Carlo metod. V obou případech je rozšíření na kontrolní problém proveden prostřednictvím myšlenky všeobecné iterace strategie (GPI). Jeden ze dvou procesů tvořících GPI řídí strategii hodnotové funkce, aby přesně předpovídala výnosy pro současnou strategii; toto je predikční problém. Jiný proces řídí strategii zaměřenou na lokální zlepšení vzhledem k aktuální

hodnotové funkci. Když je první proces založen na zkušenosti, vzniká komplikace týkající se zachování dostatečného zkoumání. TD metody lze rozdělit podle toho, zda se zabývají touto komplikací užitím přístupu on-strategie nebo off-strategie. Sarsa a actor-critic metody (viz obr. 2) jsou metody on-strategie a Q-učení⁵ a R-učení jsou metody off-strategie. Q-učení se nestará o aktuální strategii a agent funguje dobře i s náhodnou strategií. Naopak Sarsa je realističtější, protože se učí Q-funkci na základě toho, co se skutečně stalo a ne co by se mohlo stát [24].



Obrázek 2 - Architektura actor-critic. Zdroj: [24]

2.4.4 Fuzzy systémy adaptované genetickými algoritmy a symbiotickou evolucí

Genetické algoritmy (GA) patří do třídy evolučních algoritmů, které mimo ně zahrnují také evoluční programování, evoluční strategii a genetické programování. Jsou to vyhledávací algoritmy založené na mechanismu přirozeného výběru a principech genetiky. Jejich velkou výhodou je poměrná jednoduchost. Ideovým vzorem pro genetické algoritmy byly principy vývoje, které se uplatňují v přírodě. Zde existují populace jednotlivých živočišných druhů, složených z jedinců různých vlastností. Tyto vlastnosti jsou prvotně zakódovány v jejich genech, které tvoří větší celky, chromozómy. Při křížení vznikají noví jedinci, kteří mají zpravidla náhodně část genů od jednoho rodiče a část genů od rodiče

⁵ Q-učení je metoda posilovaného učení založená na uživatelské funkci pro nějakou akci, která dává očekávaný užitek provedením dané akce v daném stavu a pro pevně dané strategie. Jednou ze silných stránek Q-učení je schopnost porovnat očekávaný užitek z dostupných akcí bez nutnosti znalosti modelu prostředí. Varianta tzv. zpožděné Q-učení dosahuje mnohem lepších výsledků, a tím se ukazuje jako vhodné učení pro Markovské rozhodovací procesy.

druhého. Přitom ve zvlášť výjimečném případě může dojít k náhodné změně některého genu v chromozómu, tzv. mutaci, která může být pro další vývoj druhu příznivá nebo ne. Podle svých vlastností má každý z potomků větší nebo menší schopnost obstát v přirozeném výběru a vytvořit další generaci. Proces výběru se stále opakuje a v jeho průběhu se zlepšují genetické vlastnosti daného druhu. Tak probíhala celá evoluce v přírodě.

V teorii umělé inteligence je genetický algoritmus proces postupného vylepšování populace jedinců opakovanou aplikací genetických operátorů, který vede k evoluci takových jedinců, kteří lépe vyhovují stanoveným podmínkám než jedinci, ze kterých vznikli. Proces konverguje k situaci, kdy je populace tvořena jen těmi nejlepšími jedinci [14].

Na rozdíl od obecného evolučního algoritmu GA, který pracuje se seznamem plných řešení problémů (Pittsburský přístup), symbiotická evoluce předpokládá, že každý jedinec v populaci představuje pouze částečné řešení. Kompletní řešení je vytvořeno zkombinováním několika jednotlivců. V obecných GA, je jedinec odpovědný za celkovou vhodnost. V symbiotické evoluci, vhodnost jedince (částečné řešení) záleží na ostatních. Částečná řešení mohou být charakterizována jako specializace. Vlastnost specializace se snaží udržovat různorodost v hledání, což zabraňuje konvergenci populace. Symbiotická evoluce se zdá být rychlejší a účinnější vyhledávacím schématem pro problémy posilovaného učení než přístupy používající obecnou evoluci [14].

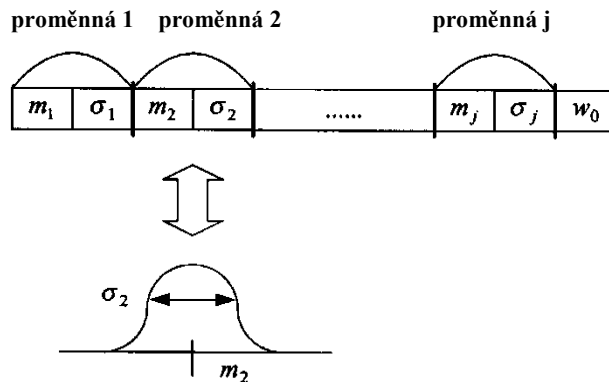
V této kapitole je představeno podrobné provedení genetického algoritmu posilovaného učení. Stejně jako tradiční GA, se tento algoritmus skládá ze tří hlavních operátorů: reprodukce, křížení a mutace. Před vysvětlením těchto tří genetických operátorů, je popsána problematika kódování, inicializace a vhodnosti přiřazení. Kódování se týká způsobu, jakým jsou funkce příslušnosti a fuzzy pravidla fuzzy systému reprezentovány jako chromozómy vhodné pro symbiotickou evoluci. Inicializace je správné přiřazení učících konstant před vstupem do evolučního procesu. Vhodnost přiřazení dává správnou fitness hodnotu každému fuzzy pravidlu během evolučního procesu. Celý algoritmus genetického posilovaného učení je popsán krok za krokem níže.

2.4.4.1 Kódování, inicializace a vhodnost přiřazení

Kódování fuzzy pravidla v chromozómu je zobrazeno na následujícím obr. 3. Gaussova funkce příslušnosti je zde používána s proměnnými m a σ reprezentujícími střed a šířku funkce příslušnosti. Konsekvant fuzzy pravidla je reprezentován konstantou (jedná se tedy o systém typu Takagi-Sugeno-Kang (TSK) nultého řádu), tj. pravidlo má obecný tvar [14]:

$$\begin{aligned}
 & \text{IF } x_1 \text{ je } \mu(m_1, \sigma_1) \text{ AND } x_2 \text{ je } \mu(m_2, \sigma_2) \text{ AND} \\
 & \dots \text{ AND } x_j \text{ je } \mu(m_j, \sigma_j) \text{ THEN } y \text{ je } w_0,
 \end{aligned}
 \tag{17}$$

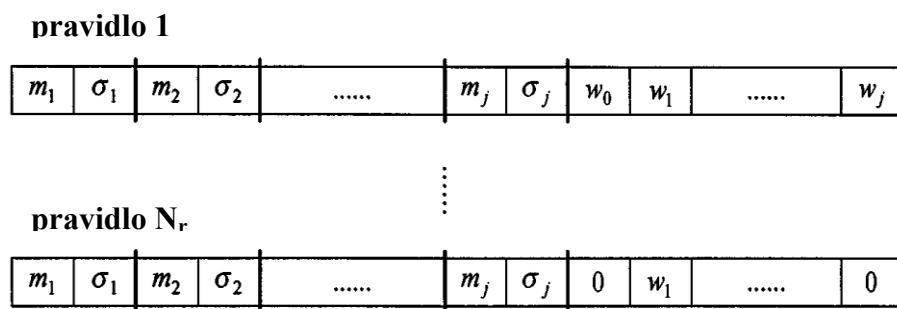
kde $\mu(m_i, \sigma_i)$ představuje Gaussovu funkci příslušnosti se středem m_i a šířkou σ_i .



Obrázek 3 - Kódování v konsekventu fuzzy pravidla s jednoprvkovou množinou v chromozomu pomocí symbiotické evoluce. Zdroj: [14]

Obr. 4 označuje TSK typ fuzzy pravidla⁶, jehož konsekventní část je lineární kombinací vstupních proměnných a konstantní hodnoty, tj. pravidlo má formu [14]:

$$\begin{aligned}
 & \text{IF } x_1 \text{ je } \mu(m_1, \sigma_1) \text{ AND } x_2 \text{ je } \mu(m_2, \sigma_2) \text{ AND} \\
 & \dots \text{ AND } x_j \text{ je } \mu(m_j, \sigma_j) \text{ THEN } y = w_0 + w_1x_1 + \dots + w_jx_j.
 \end{aligned}
 \tag{18}$$



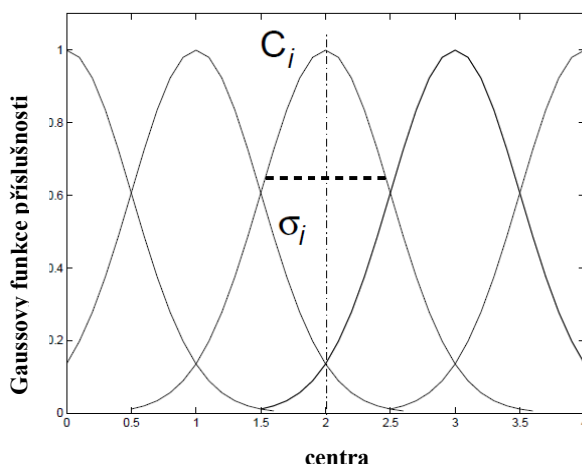
Obrázek 4 - Kódování TSK typu fuzzy pravidla v chromozomu pomocí symbiotické evoluce. Zdroj: [14]

⁶ Takagi-Sugeno-Kang (TSK) pravidlo se často využívá v řízení. Konsekventní část tohoto pravidla je funkcí vstupu, zpravidla lineární. TSK pravidlo má výhodu před konvenčními fuzzy pravidly díky jednodušším výpočtům.

Gaussova funkce příslušnosti je dána vztahem [14]:

$$\mu_{A^i}(x) = \exp\left(-\frac{(c_i-x)^2}{2\sigma_i^2}\right), \quad (19)$$

kde c_i je střed a σ_i šířka i -té fuzzy množiny A^i .



Obrázek 5 - Gaussova funkce příslušnosti. Zdroj: [14]

Inicializace. Jedinci tvořící počáteční populaci P_s by měli být nejprve generováni náhodně. Velikost populace závisí na složitosti problému. Kromě velikosti populace, je třeba upřesnit některé další parametry. Jedná se o počet fuzzy systémů N_f , které mají být vytvořeny a ohodnoceny za generaci. Počet fuzzy pravidel N_r , které tvoří fuzzy systém a pravděpodobnost mutace P_m [14].

Vhodnost přiřazení. Jak již bylo řečeno pro symbiotické evoluce, fitness hodnota pravidla (jednotlivce), je vypočtena sečtením fitness hodnot všech možných kombinací tohoto pravidla se všemi dalšími $N_r - 1$ náhodně vybranými pravidly, a poté vydělením celkovým počtem kombinací. Podrobný postup pro přiřazení fitness hodnoty je popsán následovně [14]:

Krok 1) Náhodně zvol N_r fuzzy pravidel z populace P_s pravidel, a vytvoř z těchto N_r pravidel fuzzy systém.

Krok 2) Ohodnot' provedení kombinovaného fuzzy systému pro získání fitness hodnoty.

Krok 3) Rozděl fitness hodnotu podle N_r a nahromad' rozdělenou fitness hodnotu do fitness záznamu N_r vybraných pravidel s jejich fitness záznamy nastavenými na počátku na nulu.

Krok 4) Opakuj N_f krát výše uvedený postup dokud každé pravidlo nebude zvoleno dostatečně krát a zaznamenej počet fuzzy systémů, ve kterých se každý jednotlivec podílí.

Krok 5) Rozděl kumulovanou fitness hodnotu každého jednotlivce podle počtu, kolikrát byl vybrán.

Průměrná fitness hodnota představuje provedení pravidla. Pro TSK typ fuzzy pravidla, po výpočtu průměrné fitness hodnoty každého pravidla, je tato hodnota je dále dělená K + počtem parametrů v pravidle, kde K je konstanta, takže pravidlu s méně parametry je přiřazena vyšší fitness hodnota. Tímto způsobem, jsou vybrány pouze významné výrazy a použity v konsekventu TSK typu fuzzy pravidla. V této kapitole, je síla fuzzy pravidla vypočtená provedením následujících AND operací na pravdivých hodnotách každé proměnné k jeho odpovídajícím fuzzy množinám [14]:

$$\begin{aligned}\phi &= \prod_{i=1}^j \exp\left\{-\frac{(x_i - m_i)^2}{\sigma_i^2}\right\} \\ &= \exp\left\{-\sum_{i=1}^j \frac{(x_i - m_i)^2}{\sigma_i^2}\right\},\end{aligned}\quad (20)$$

výstup fuzzy systému je vypočítán [14]:

$$y^* = \frac{\sum_{i=1}^{Nr} \phi_i y_i}{\sum_{i=1}^{Nr} \phi_i}, \quad (21)$$

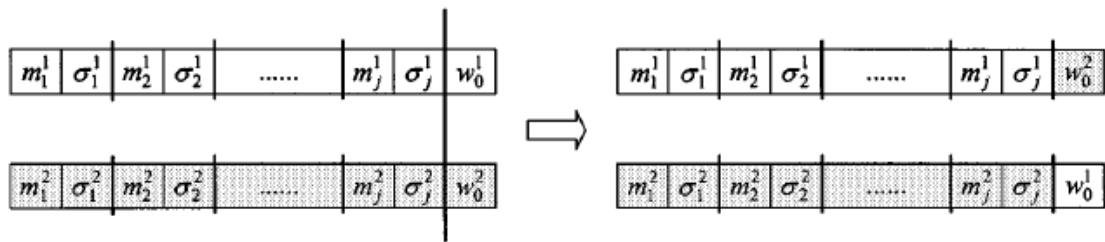
kde y_i je řídicí akce doporučená i -tým pravidlem.

2.4.4.2 Reprodukce, křížení a mutace

Reprodukce [14] je proces, ve kterém jsou jednotlivé řetězce kopírovány podle jejich fitness hodnoty. Před provedením procesu je populace nejprve tříděna podle fitness hodnoty každého jednotlivce. Horní polovina nejlepších provedení jednotlivců v populaci postoupí do příští generace. Zbývající polovina bude vygenerována provedením operace křížení na jednotlivcích v horní polovině generace předchůdců.

Reprodukce řídí vyhledávání směrem k nejlepším existujícím jednotlivcům, ale nevytváří žádné nové jedince. Nové jednotlivci jsou vytvořeny především operací *křížení*. Aby bylo možné vybrat jednotlivce pro křížení, výběr se provádí na horní polovině nejlepších provedení jednotlivců. Při výběru dvou nebo více jednotlivců v horní polovině populace jsou vybrány

náhodně, a jejich fitness hodnoty jsou porovnány. Jedinec s nejvyšší fitness hodnotou je vybrán jako jeden z předchůdců. Další předchůdce je vybrán stejným způsobem. Potomci jsou vytvořeni provedením křížení na vybrané předchůdce. Některé z běžně používaných technik křížení jsou jedno bodové křížení, dvou bodové křížení, vícenásobné bodové křížení, křížení náhodnou výměnou a rovnoměrné křížení. V tomto případě, je použito jedno bodové křížení [14]. Na obr. 6 jsou nový jedinci vytvořeni výměnou konsekventních částí mateřských pravidel. Po operaci, budou jedinci s nízkou výkonností nahrazeni nově produkovanými potomky.



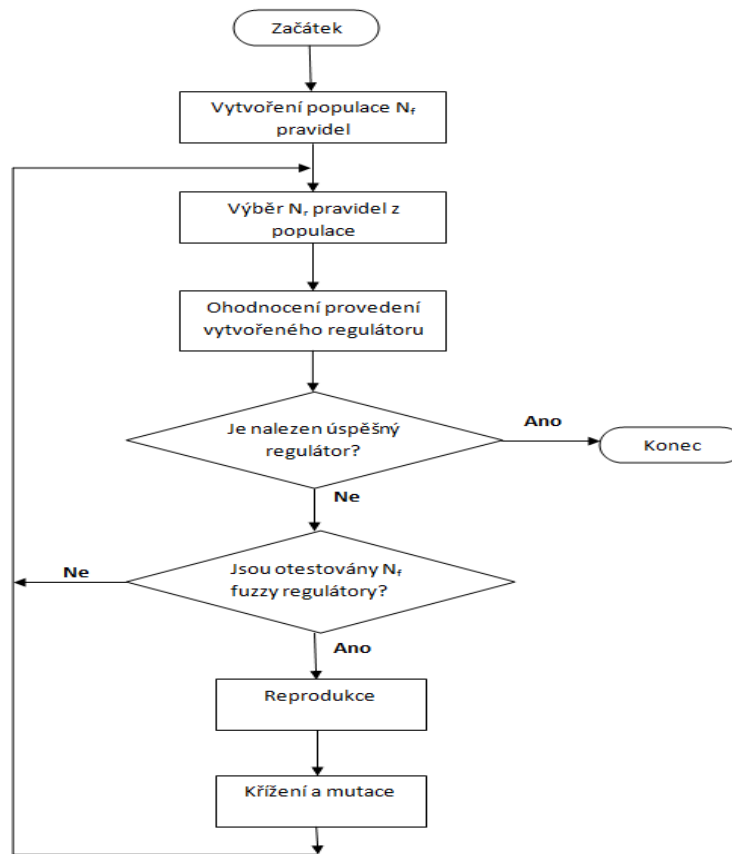
Obrázek 6 - Operace křížení pravidel (jednotlivců) v konsekventní části pravidla. Zdroj: [14]

Mutate je operátor, kdy je alela⁷ genu náhodně změněna. S mutací, mohou být zavedeny nové genetické materiály do populace. Mutace by měla být použita šetrně, protože je to náhodný vyhledávací operátor, jinak s vysokou mírou mutace, se algoritmus stane trochu víc než jen náhodné vyhledávání, proto se užívá pravděpodobnost mutace $P_m=0.01$.

Tento výše uvedený proces je prováděn opakovaně, a je ukončen, jakmile je dosaženo předem stanoveného stavu. Blokované schéma návrhu metody SEFC (Symbiotic Evolution based Fuzzy Controller)⁸ je znázorněno na následujícím obr. 7.

⁷ Alela je konkrétní forma genu. Existují většinou různé formy, zodpovědné za různé projevy téhož genu. V rámci 1 organismu jsou 2 alely pro 1 gen.

⁸ Genetický algoritmus posilovaného učení pro návrh fuzzy systému na základě symbiotické evoluce.



Obrázek 7 - Blokové schéma navrhované metody konstrukce SEFC. Zdroj: [14]

Nový genetický algoritmus posilovaného učení, tzv. návrh SEFC metody, byl navržen pro tvorbu fuzzy regulátoru ve [14]. Ve srovnání s tradičními GA založenými na fuzzy metodě návrhu regulátoru, navrhovaná metoda SEFC vyžaduje méně pokusů a méně procesorového času. Vstupní prostor SEFC je rozdělen flexibilním způsobem. Pravidla používaná v SEFC mohou být několika typů, např. TSK typ fuzzy pravidla.

2.5 Dílčí závěr

Každá třída uvedených metod má své silné a slabé stránky. Metody dynamického programování jsou dobře vyvinuté matematicky, ale vyžadují komplexní a přesný model prostředí. Monte Carlo metody nevyžadují model a jsou koncepčně jednoduché, ale nejsou vhodné pro přírůstkové počítání krok za krokem. Temporal Difference metody nevyžadují žádný model a jsou zcela přírůstkové, ale jsou složitější k analyzování. A poslední představenou metodou je efektivní genetický algoritmus posilovaného učení pro návrh fuzzy systému na základě symbiotické evoluce. Metody se také liší v několika směrech s ohledem na jejich účinnosti a rychlosti konvergence.

3 Předzpracování finanční časové řady

Tato kapitola je věnována předzpracování finanční časové řady pomocí indikátorů technické analýzy a její přípravě pro predikci akciových kurzů. Pro správné řešení zadaného úkolu bylo zapotřebí získat vhodná data, tedy data dostatečná v počtu a kvalitě. Pro práci byl jako časová řada zvolen vývoj indexu Nasdaq Composite od začátku roku 2007 do konce roku 2010 uváděný v bodových hodnotách v pracovních dnech [18].

3.1 Zdrojová data

Jako vstupní data byla použita finanční časová řada o délce 4 roky obsahující hodnoty akciového indexu Nasdaq Composite (symbol IXIC) za pracovní dny – viz graf 1. Tento index se stal vůbec nejsledovanějším díky svému širokému záběru měřicímu vývoj všech akcií obchodovaných na The Nasdaq Stock Market, a to jak amerických, tak i neamerických. Jeho komponenty je v současné době přes tři tisíce akciových titulů. Jde tedy o index souhrnný. Byl zaveden roku 1971 krátce po vzniku Nasdaq. Přestože je podobně jako celý Nasdaq zaměřen především na technologické akcie, obsahuje také akcie společností z odvětví finančního a spotřebního průmyslu, biotechnologií i akcie průmyslových podniků. Výhodou indexu Nasdaq Composite je velký počet akcií, nevýhodou je vyšší zaměření na technologický sektor. I přesto představuje tento index více než jen vzorek technologického odvětví.

Nasdaq Composite je vážený index - vahou jednotlivých komponentů je přitom jejich tržní kapitalizace, tedy součin kurzu akcie a celkového počtu vydaných akcií. To v praxi znamená, že daný titul ovlivní vývoj celého tohoto indexu jen takovou vahou, jakou má v celkové tržní kapitalizaci trhu Nasdaq.

Počáteční hodnota indexu byla v roce 1971 stanovena na 100 bodů, v březnu 2000 dosáhl index rekordu 5048 bodů, v srpnu 2002 prudce poklesl pod 1300 bodů. Na začátku roku 2007 dosáhl index hodnoty 2423 a na konci roku 2010 hodnoty 2642. Zhodnocení za sledované období 2007 až 2010 je tedy kolem 9 %.

Mimo index Nasdaq Composite je ještě užíván index Nasdaq 100, který v sobě zahrnuje 100 nejobchodovanějších titulů na Nasdaqu. Na tento index jsou vázány mnohé deriváty.



Graf 1 - Vývoj hodnot indexu Nasdaq composite. Zdroj: [vlastní]

3.2 Předpracování časové řady

Závěrečné⁹, nejvyšší, nejnižší¹⁰ a otevírací¹¹ ceny akcie lze předpracovat pomocí indikátorů technické analýzy (TA). TA se zabývá studiem historických údajů o obchodování s cílem předpovědět krátkodobé pohyby cen cenných papírů. V TA se předpokládá, že všechny cenotvorné informace o pohybu cen jsou už obsažené v minulých údajích. Základní vstupní informace o trhu a chování cenných papírů v minulosti může poskytnout informace o budoucím chování cen akcií [20].

V TA se využívají dva druhy metod analýzy údajů. Jedná se o metody založené na analýze grafů a metody využívající indikátory TA. Grafické metody jsou založené na odhalování pravidelně se opakujících formací. Na jejím základě lze předpovídat další vývoj kurzu cenného papíru. Indikátory TA jsou založené na využití statistických metod. Snahou obou

⁹ Pod závěrečnou cenou se rozumí cena, za kterou se uskutečnil poslední obchod s danou akcií ve stanovený obchodovací den.

¹⁰ Nejvyšší (nejnižší) cena akcie je nejvyšší (nejnižší) hodnota ceny, která byla dosažena v daný obchodovací den s danou akcií

¹¹ Pod otevírací cenou se rozumí cena, za kterou se uskutečnil první obchod s danou akcií v daný obchodovací den.

metod je najít a identifikovat nákupní a prodejní signály na trhu. Nákupní signál předpovídá budoucí růst ceny akcie a reakcí na ní by měla být koupě dané akcie. Prodejní signál předpovídá budoucí pokles ceny akcie a reakcí na ní by měl být prodej analyzované akcie [20].

Pro předzpracování jsou použity následující indikátory:

- Indikátor založený na výpočtu jednoduchého klouzavého průměru (Simple Moving Average – SMA);
- Indikátor založený na výpočtu váženého klouzavého průměru (Weighted Moving Average – WMA);
- Indikátor momentum (Momentum – MOM);
- Směrodatná odchylka (Standard Deviation – STD).

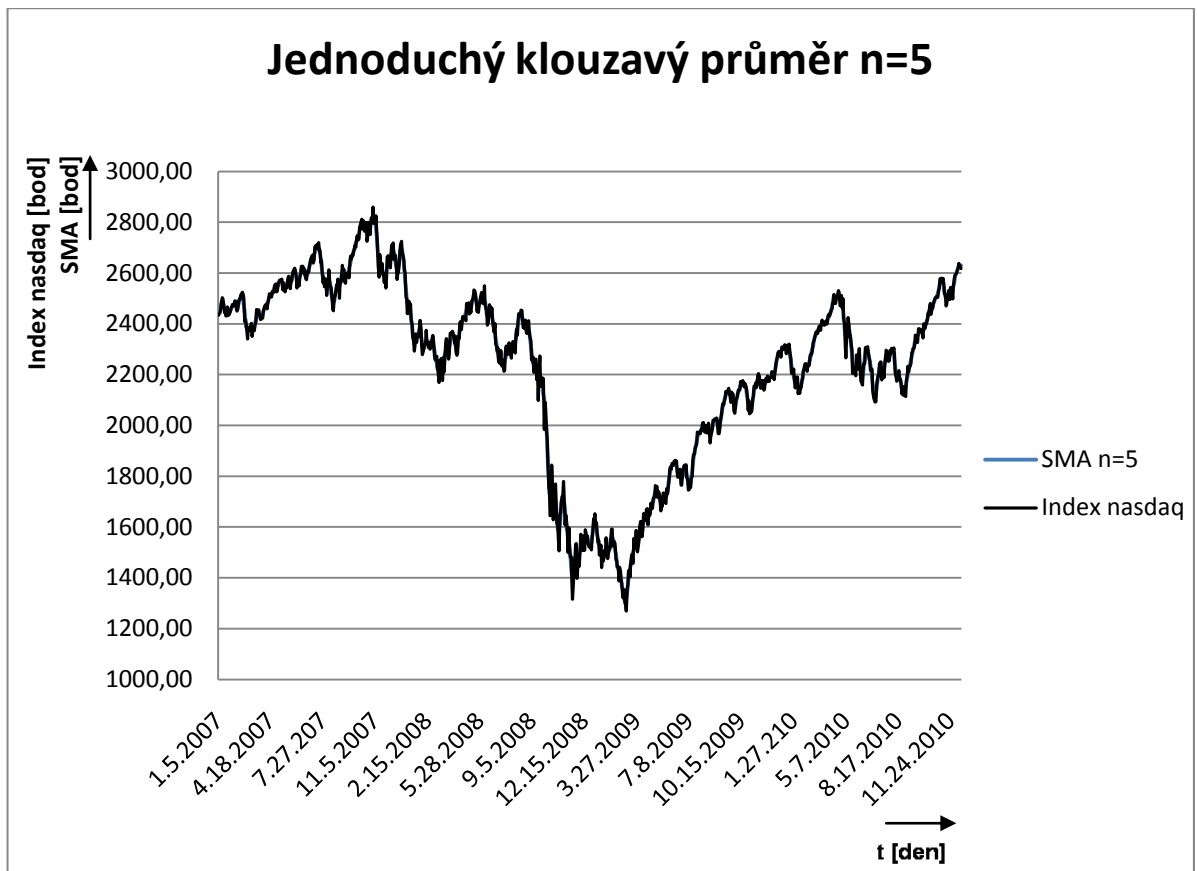
Při výpočtu vybraných indikátorů technické analýzy byla zvolena perioda $n = 5$ dní z důvodu zveřejňování hodnot indexu pouze v pracovních dnech. Zároveň je tak možné vzít v úvahu anomálie, ke kterým dochází vždy na začátku týdne (pokles akciového indexu). Při předzpracování časové řady byly dále otestovány periody o délce $n = 3$ a $n = 7$, viz příloha 3. Potvrdilo se, že perioda $n = 5$ nejlépe modeluje průběh akciového indexu.

3.2.1 Jednoduchý klouzavý průměr

Jednoduchý klouzavý průměr (SMA) [20] slouží k potlačení velkých výkyvů na křivce. Počítá průměrnou hodnotu za danou volitelnou periodu (čím větší je zvolena perioda, tím více je potlačena nerovnost křivky a přiblížena k aproximovanému vývoji). Existuje několik typů MA, např.: jednoduchý, exponenciální, trojúhelníkový, proměnlivý, váhový. Jednoduchý MA přiřazuje stejnou váhu všem hodnotám (cenám). Při řešení zadaného úkolu byla zvolena perioda n rovno 5 dní, graf 2. MA popisuje následující vzorec [20]:

$$SMA_{t,n} = \frac{1}{n} \sum_{t=1}^n C_{t,n}, \quad (22)$$

kde n je délka periody a $C_{t,n}$ je hodnota akciového indexu v čase t za periodu o délce n .



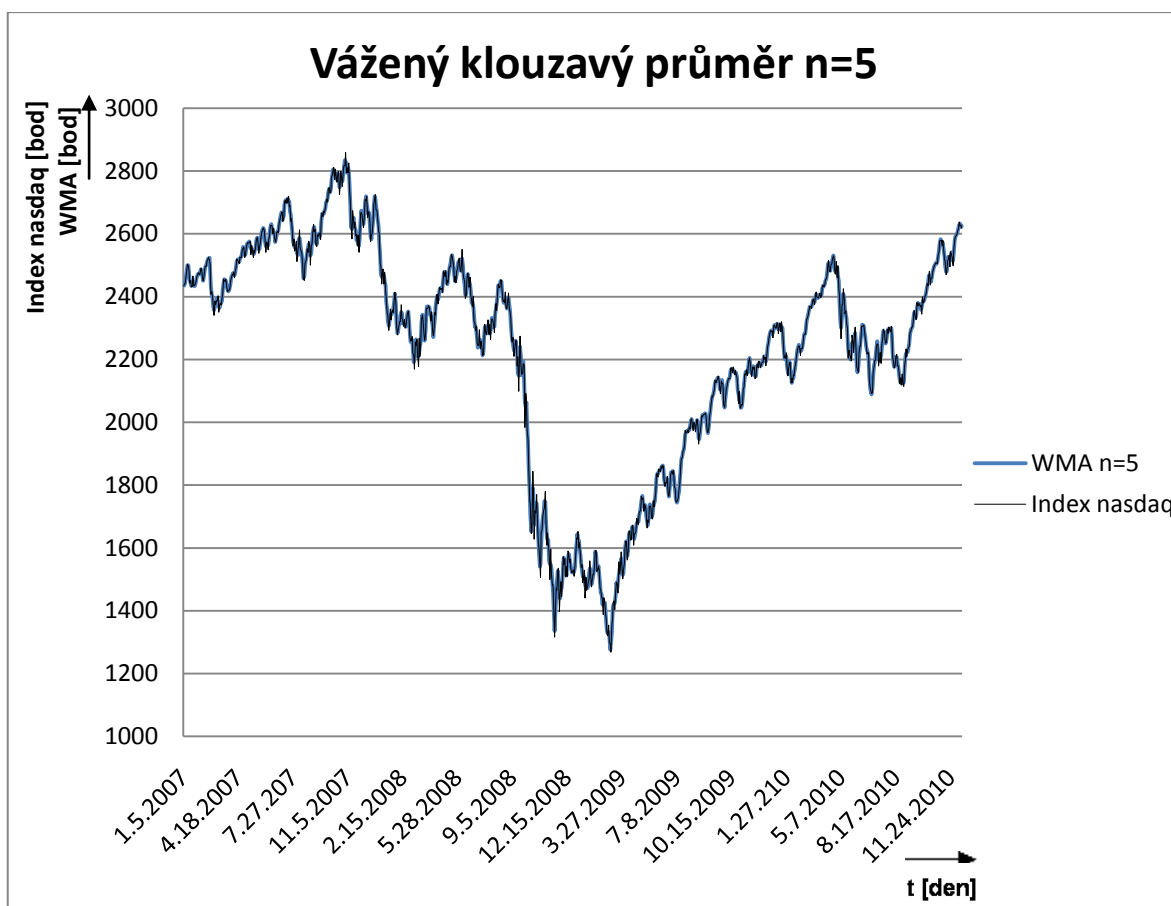
Graf 2 - Klouzavý průměr n=5. Zdroj: [vlastní]

3.2.2 Vážený klouzavý průměr

Vážený klouzavý průměr [20] se snaží odstranit nedostatek jednoduchého klouzavého průměru, ve kterém má každá cena stejnou důležitost pro výslednou hodnotu průměru. Při výpočtu váženého klouzavého průměru přidělujeme každé kurzovní hodnotě váhu úměrnou stáří kurzu. Nejmenší váhu (zpravidla 1) má nejstarší kurz, druhý nejstarší kurz potom má váhu 2, atd. Tímto způsobem zvyšujeme váhu až do hodnoty časové periody. Vážené klouzavé průměry zobecňují jednoduché klouzavé průměry tím, že průměrovaným hodnotám nedávají stejné váhy. Váhy vhodným způsobem symetricky klesají od středu okna k jeho oběma krajům. WMA pro $n = 5$ (graf 3) se vypočítá podle vztahu [20]:

$$WMA_{t,n} = \frac{1}{35} (-3C_{t-2} + 12C_{t-1} + 17C_t + 12C_{t+1} - 3C_{t+2}), \quad (23)$$

kde C_t je hodnota akciového indexu v čase t .



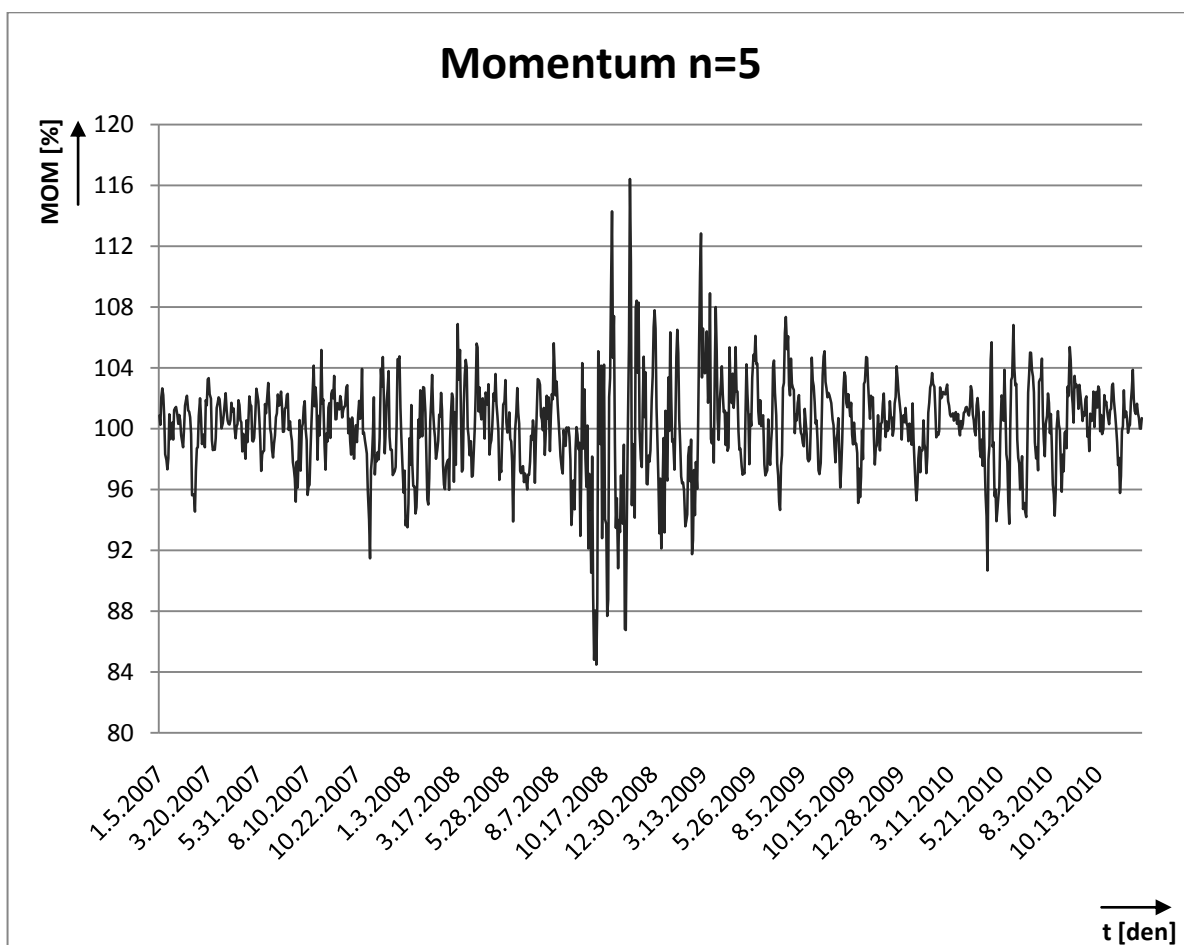
Graf 3 - Vážený klouzavý průměr n=5. Zdroj: [vlastní]

3.2.3 Indikátor momentum

Momentum [20] je indikátor, který je vytvořen ke sledování a identifikaci cyklického kolísání a krátkodobých trendů ve vývoji akciových kurzů. Pomáhá analyzovat jak cenovou úroveň, tak intenzitu oscilace kurzů (graf 4). Momentum se hlavně používá pro odhad změny trendu. Zobrazuje vývoj a změny vývoje přenesené do pásma hodnot. Zobrazené výsledky jsou procentuální odchylky od bodů vzdálených od daného bodu o volenou periodu n , která je v řešeném případě pět. Hodnoty MOM oscilují kolem hodnoty 100 [%]. Vyšší n způsobuje hladší průběh vývoje tohoto indikátoru. Jeho význam spočívá v tom, že umožňuje předpovídat změny trendu ve vývoji ceny dříve, než se tyto změny projeví ve vývoji samotné ceny. Vypočítá se jako [20]:

$$MOM_t = (C_t - C_{t-n}) * 100, \quad (24)$$

kde C_t je hodnota akciového indexu v čase t a C_{t-n} je jeho hodnota před n dny.



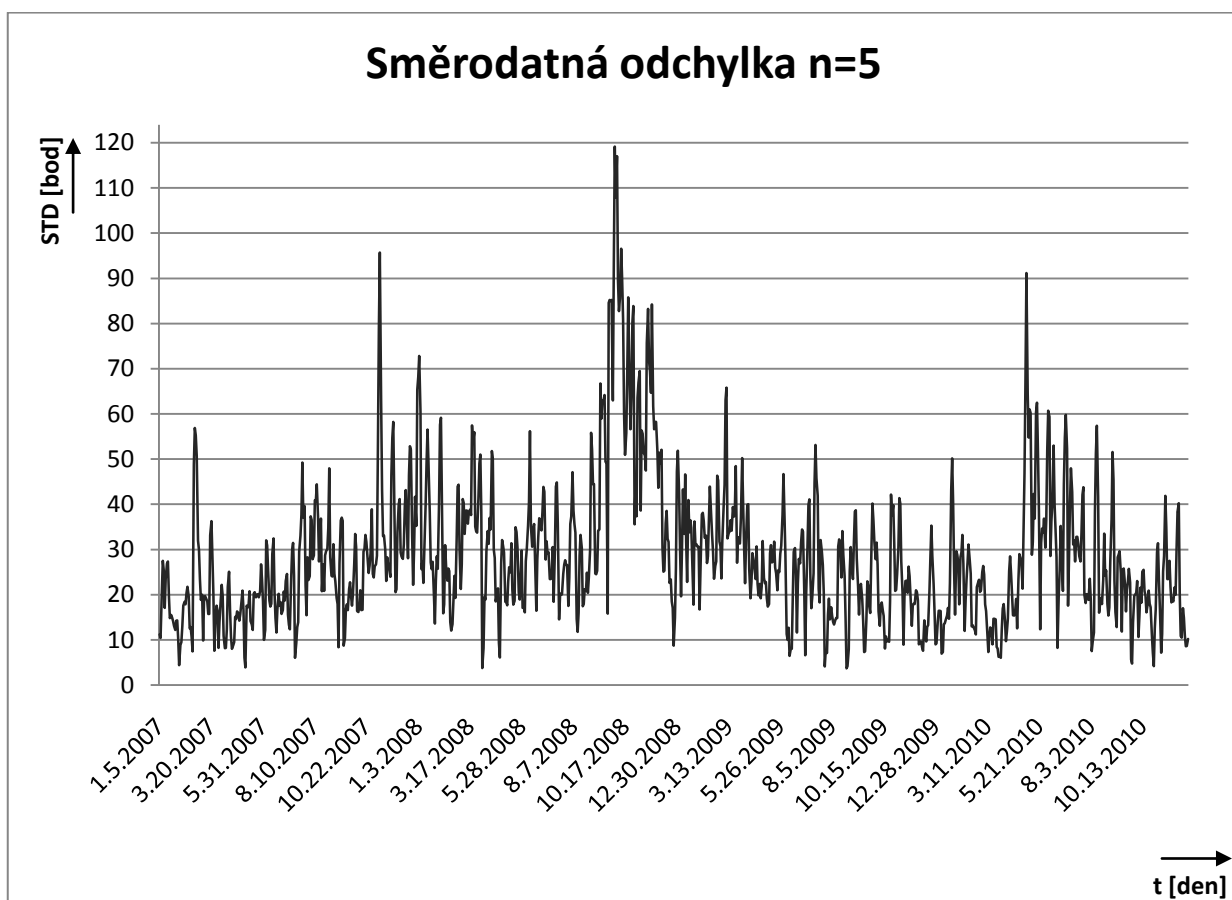
Graf 4 – Indikátor momentum n=5. Zdroj: [vlastní]

3.2.4 Směrodatná odchylka

Směrodatná odchylka [20] je statistickou mírou cenových výkyvů. STD základního souboru určená z náhodného výběru vyjadřuje, jak se hodnoty liší od průměrné hodnoty (graf 5). Zhruba řečeno vypovídá o tom, jak moc se od sebe navzájem liší typické případy v souboru zkoumaných čísel. Vysoké hodnoty STD se vyskytují v případech, kde se analyzovaná veličina podstatně mění. Podobně nízké hodnoty se objevují v případech, kdy je sledovaná veličina stabilní. Při výpočtu se používá jednoduchý MA s délkou periody n [20]:

$$STD_{t,n} = \sqrt{\frac{\sum_{i=1}^n (C_{t,n} - SMA(C_{t,n}))^2}{n}}, \quad (25)$$

kde n je délka periody SMA, $C_{t,n}$ je hodnota akciového indexu v čase t za periodu o délce n a $MA(C_{t,n})$ jednoduchý klouzavý průměr z $C_{t,n}$.



Graf 5 - Směrodatná odchylka n=5. Zdroj: [vlastní]

3.3 Dílčí závěr

Pro předzpracování časové řady bylo využito indikátorů technické analýzy jednoduchého klouzavého průměru, váženého klouzavého průměru, směrodatné odchylky a momentum. Zvolená perioda $n = 5$ dní potvrdila, že nejlépe modeluje průběh akciového indexu.

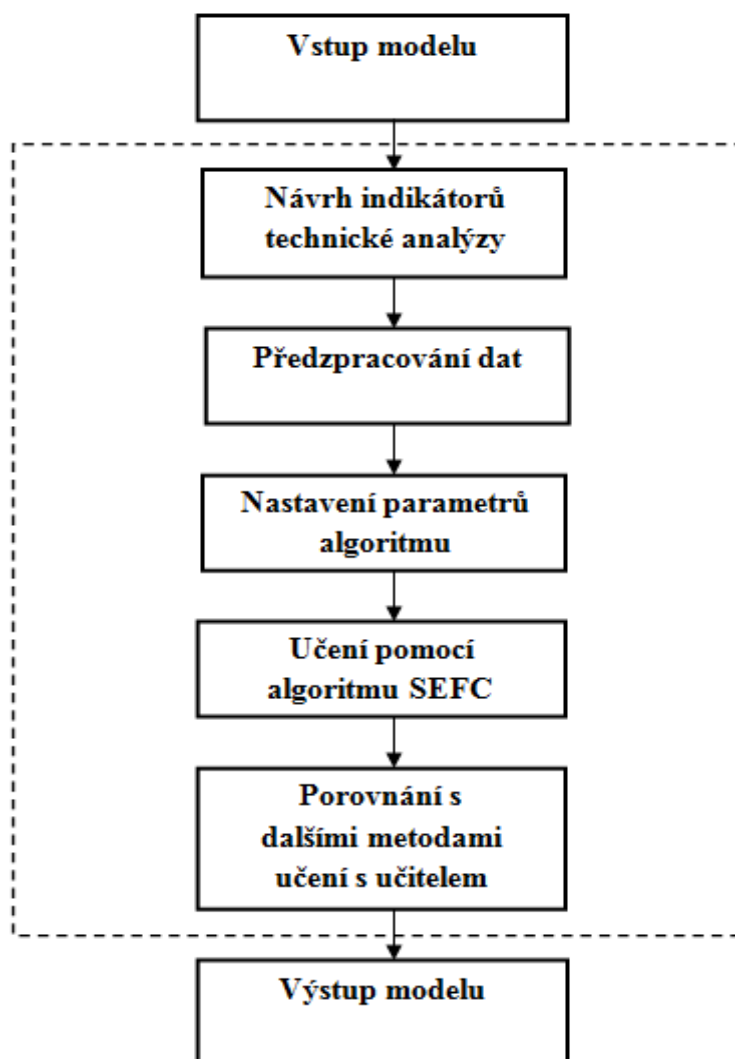
Jestliže jsou proměnné ve stejných jednotkách a souměřitelné, lze predikovat přímo z hrubých, netransformovaných dat. Standardizace dat se často doporučuje, když jsou proměnné v různých jednotkách. Jelikož rozdíl v hodnotách mezi jednotlivými sloupci je značný, bylo nutné provést normalizaci dat. Nyní jsou data připravena pro predikci vývoje.

4 Návrh modelu pro predikci akciových kurzů

Cílem této kapitoly je navrhnout takový model, který by mohl být využit pro predikci akciových kurzů. Návrh modelu bude sestaven na základě metody posilovaného učení. Výstupem této kapitoly je navržený model, který bude dále realizován v programovém prostředí Keel, které je volně dostupné. Další součástí této kapitoly je formulace parametrů metody genetického posilovaného učení na základě symbiotické evoluce, charakteristika vstupních dat a jejich následné rozdělení na trénovací a testovací množinu.

4.1 Návrhu modelu

Postup při návrhu modelu pro predikci akciových kurzů je znázorněn na obr 8.



Obrázek 8 - Návrh modelu pro predikci akciových kurzů. Zdroj: [vlastní]

Návrh modelu pro predikci akciových kurzů je založen na bázi posilovaného učení. Model je složen z návrhu indikátorů technické analýzy, předzpracování dat, nastavení parametrů SEFC a učení pomocí této metody. Na závěr budou výsledky této metody porovnány s výsledky neuronových sítí s učením s učitelem.

Vstup do modelu je představován finanční časovou řadou akciového indexu Nasdaq Composite. Ukázka dat je uvedena v příloze 1. Fáze předzpracování dat zahrnuje několik dílčích částí. Mezi nejdůležitější části této fáze patří normalizace dat a jejich následné rozdělení na testovací a trénovací množinu. Pak dochází k samotnému učení na bázi posilovaného učení. Toto učení je provedeno pomocí algoritmu genetického posilovaného učení na základě symbiotické evoluce popsaného výše. Výsledkem učení je model predikce akciových kurzů s nejvyšší přesností.

4.2 Transformace dat

Před započítím prvních kroků předzpracování dat, je dobré pro lepší orientaci v datech zpracovat popisnou statistiku, která podává stručný přehled o datech v souboru. Z tohoto důvodu byly vytvořeny jednotlivé popisné statistiky, které jsou znázorněny v tabulkách v příloze 2. Hodnoty vstupních parametrů se výrazně liší. Tato skutečnost může způsobovat, že určité znaky (proměnné) se jeví jako dominantní. Z tohoto důvodu je vhodné data upravit tak, aby všechny proměnné měly stejnou normu (nejlépe jednotkovou), čehož je možné dosáhnout pomocí normalizace. Normalizované hodnoty byly vypočteny v MS Excel pomocí vztahu:

$$n_{ij} = \frac{x_{ij}}{n_i}, \quad (26)$$

$$n_i = \left[\sum_{j=1}^p (x_{ij})^2 \right]^{\frac{1}{2}}, \quad (27)$$

kde n_{ij} vyjadřuje normalizovanou hodnotu pro i -tý objekt a j -tou proměnnou, x_{ij} představuje původní hodnotu pro i -tý objekt a j -tou proměnnou a n_i je norma j -té proměnné.

4.3 Rozdělení dat na trénovací a testovací množinu

Jeden z nejdůležitějších kroků předzpracování je rozklad množiny dat A na trénovací množinu A_{train} , což je množina uspořádaných hodnot, která se používá ve fázi učení, a testovací množinu A_{test} , což je množina uspořádaných hodnot, která se používá ve fázi života, za účelem otestování získaných znalostí v průběhu učení. Důležitost reprezentativnosti trénovacích dat je mimořádná, neboť znalosti získané při učení z těchto dat musejí být aplikovatelné na testovací data, která nejsou v průběhu učení použita. Nevhodně vybraná trénovací data negativně ovlivňují kvalitu učení. Proces učení tedy spočívá v tom, že na vstupy sítě se opakovaně připojuje soubor vzorů požadovaného chování, tzv. trénovací množina A_{train} a hledá se optimální nastavení parametrů prediktoru, při kterém definovaná funkce rozdílu předpokládaného výstupu a výstupu prediktoru – chyba E , bude minimální [20].

Množinu objektů A je třeba rozdělit na disjunktní podmnožiny:

$$A = C_1 \cup C_2 \cup \dots \cup C_p, \quad (28)$$

kde i -tá podmnožina C_i obsahuje n_i objektů z množiny A

$$C_i = \{o_1^{(i)}, o_2^{(i)}, \dots, o_{n_i}^{(i)}\} \subset A, \quad (29)$$

kde $i = 1, 2, \dots, p$.

Podstatnou roli v návrhu modelu hrálo tedy rozdělení vstupní množiny dat na množinu testovací a trénovací. Pomocí trénovacích dat se algoritmus SEFC učí, na testovacích datech se poté testuje. V případě navrženého modelu bylo využito 10 násobné křížové validace, tj. $p = 10$. Při testování metodou křížová validace se data dopředu rozdělí na 10 částí tak, že vždy jedna desetina se vyjme pro testování a zbylých devět desetin se použije pro učení. Celý tento postup se zopakuje desetkrát a výsledek testování se zprůměruje. Proto se tomuto způsobu testování říká desetinásobná křížová validace. Toto rozdělení bylo zvoleno na základě předchozích pokusů nastavení, jelikož dávalo nejlepší výsledky.

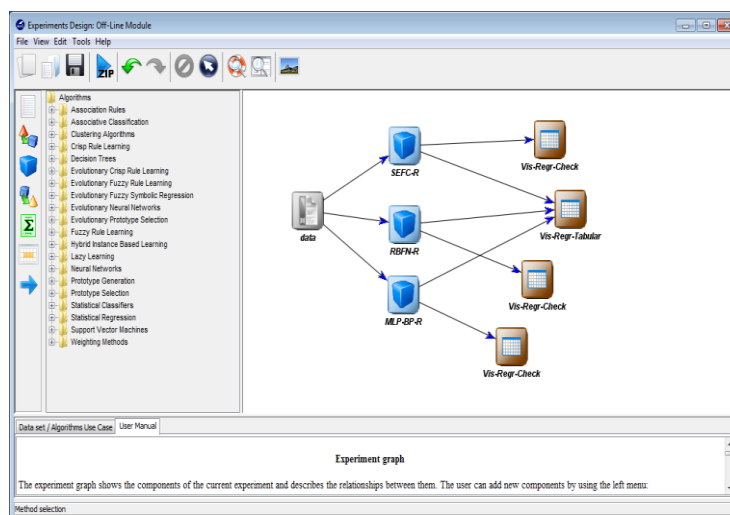
4.4 Algoritmus fuzzy symbiotické evoluce

Pro návrh modelu predikce akciových kurzů bude využito metody posilovaného učení. Jako algoritmus pro predikci byl vybrán algoritmus fuzzy symbiotické evoluce (Symbiotic Evolution based Fuzzy Controller design method - SEFC).

C. F. Juang a kol. [14] navrhli tento efektivní genetický algoritmus posilovaného učení pro návrh fuzzy systému na základě symbiotické evoluce, který při použití návrhu fuzzy regulátoru, doplňuje místní mapování vlastnosti fuzzy pravidla. Volné parametry střed a šířka funkce příslušnosti a fuzzy pravidla jsou všechny laděny automaticky pomocí symbiotické evoluce. Pokud jde o výstup z pravidel, pouze významné vstupní proměnné jsou vybrány k účasti v konsekventu pravidla. Podrobný popis tohoto algoritmu je představen v kapitole 2.4.4. Mezi nejdůležitější parametry algoritmu SEFC, které ovlivňují kvalitu výsledků, patří následující parametry:

- *Velikost populace*, která nabývá hodnot 10 až 10000,
- *Počet fuzzy pravidel*, tvořících fuzzy systém, nabývá hodnot 10 až 10000,
- *Počet fuzzy systémů*, které mají být vytvořeny a hodnoceny za generaci, nabývá hodnot 10 až 10000,
- *Parametr K*, který nabývá hodnot 0 až 1,
- *Pravděpodobnost mutace*, která nabývá hodnot 0 až 1.

V programovém prostředí Keel byly realizovány modely predikce akciových kurzů pomocí algoritmu SEFC a pomocí metod učení s učitelem. Obr. 9 prezentuje realizaci modelu ve zvoleném prostředí.



Obrázek 9 - Návrh modelu v prostředí Keel. Zdroj: [vlastní]

4.5 Dílčí závěr

Tato kapitola byla zaměřena na návrh modelu predikce akciových kurzů a následně jeho realizaci ve zvoleném programovém prostředí Keel. Bylo provedeno předzpracování vstupních dat a jejich následné rozdělení na trénovací a testovací množinu. Byly definovány parametry algoritmu SEFC, na základě kterých budou dále provedeny experimenty.

5 Zhodnocení výsledků experimentů

Samotná predikce finanční časové řady je realizována v programovém prostředí Keel. Veškeré experimenty s navrženým modelem byly provedeny v tomto programovém prostředí, které je volně dostupné. Cílem kapitoly je testovat jednotlivé navržené modely predikce na základě změn výše uvedených parametrů algoritmu SEFC. Následuje zhodnocení výstupů a pro porovnání výsledků byly použity metody učení s učitelem, dopředná neuronová síť MLP-BP a neuronová síť typu RBF.

5.1 Nastavení parametrů algoritmu

Mezi nejdůležitější parametry algoritmu SEFC, které ovlivňují výsledky predikce, patří počet fuzzy pravidel tvořících fuzzy systém a počet fuzzy systémů vytvořených a hodnocených za generaci. Mezi další parametry, které mohou ovlivnit výsledky, patří pravděpodobnost P_m , že gen bude vystavený mutaci. Hodnota P_m by měla být volena nízká, protože se jedná o náhodný vyhledávací operátor a s vysokými hodnotami se algoritmus stane jen o něco více než náhodným vyhledáváním. Parametr K je konstanta, která ovlivňuje počet parametrů ve fuzzy pravidlech a velikost populace P_s závisící na složitosti realizovaného problému. Výchozí hodnoty těchto parametrů jsou shrnuty v tabulce 1.

Tabulka 1- Výchozí nastavení parametrů algoritmu SEFC. Zdroj: [vlastní]

Výchozí nastavení parametrů algoritmu SEFC		
P_s	Velikost populace	100
N_r	Počet fuzzy pravidel, které tvoří fuzzy systém	50
N_f	Počet fuzzy systémů, které mají být tvořeny a hodnoceny za generaci	100
K	Počet parametrů ve fuzzy pravidlech	30
P_m	Pravděpodobnost mutace	0.01

5.2 Zhodnocení výsledků algoritmu

Pro dosažení co nejlepších výsledků byly následně měněny parametry, které mohou ovlivnit kvalitu navrženého modelu. Pro zhodnocení výsledků algoritmu je použita střední kvadratická odchylka (Mean Square Error, MSE) měřící přesnost předpovědi jako průměr druhých mocnin odchylek vypočtených a skutečných hodnot a směrodatná odchylka (Standard Deviation, STD). Směrodatná odchylka měří volatilitu (proměnlivost, kolísavost) dané proměnné (v tomto případě MSE). Volatilita měřená směrodatnou odchylkou je ve finančním světě často ztotožňována s rizikem daného finančního instrumentu. Čím vyšší volatilita, tím

vyšší pravděpodobnost, že vývoj kurzu daného instrumentu půjde opačným směrem, než se předpokládá. Směrodatnou odchylku lze vypočítat jakou druhou odmocninou z rozptylu (Variance). Rozptyl pak jako součet druhých mocnin rozdílu predikované ceny instrumentu v daném časovém okamžiku a průměru za danou časovou periodu a tento součet se vydělí velikostí dané časové periody (počtem jednotlivých hodnot v dané periodě). STD lze vyjádřit vztahem [20]:

$$STD = \sqrt{\frac{1}{N} \sum_{i=1}^n (x_i - \bar{x})^2}, \quad (30)$$

kde N je velikost sledovaného časového úseku, x_i je i -tá hodnota (cena) instrumentu z N hodnot v daném časovém úseku a \bar{x} je průměr hodnot v daném časovém úseku. Důsledkem nelinearity je skutečnost, že větší odchylky u MSE jsou více penalizovány než v lineárním případě průměrné absolutní odchylky (MAD), kde se měří přesnost předpovědi jako průměr z absolutních odchylek vypočtených a skutečných hodnot. MSE se vypočte jako [20]:

$$MSE = \frac{\sum (x(t) - \hat{x}(t))^2}{N} = \frac{\sum e^2(t)}{N}, \quad (31)$$

kde N je celkový počet prvků v množině výstupů, $x(t)$ je skutečná hodnota a $\hat{x}(t)$ je hodnota predikovaná.

Bylo provedeno několik experimentů s různým nastavením parametrů. Pro různý počet N_f fuzzy systémů při zachování výchozích hodnot ostatních parametrů bylo dosaženo následujících výsledků. Z uvedené tabulky je patrné, že pro $N_f = 200$ fuzzy systémů je MSE a STD nejmenší.

Tabulka 2 - Výsledky genetického algoritmu SEFC pro různé hodnoty N_f . Zdroj: [vlastní]

Počet fuzzy systémů N_f	Trénovací data		Testovací data	
	MSE	STD	MSE	STD
50	2593.264	2375.293	2591.599	2257.281
100	2389.711	3019.254	2391.296	3312.419
150	2463.710	1417.599	2465.288	2100.898
200	2310.720	936.851	2311.550	1415.567
250	2418.755	1679.619	2417.533	1494.505
300	2466.095	2565.599	2467.628	2897.878

Dalším důležitým parametrem je počet fuzzy pravidel N_r tvořících fuzzy systém. Při zachování ostatních parametrů na výchozí hodnotě bylo dosaženo nejlepších výsledků při $N_r = 60$ fuzzy pravidlech.

Tabulka 3 - Výsledky genetického algoritmu SEFC pro různé hodnoty N_r . Zdroj: [vlastní]

Počet fuzzy pravidel N_r	Trénovací data		Testovací data	
	MSE	STD	MSE	STD
10	2429.414	1111.156	2430.262	1339.076
20	2561.180	1878.206	2559.678	1672.719
30	2508.585	2994.893	2502.962	2296.406
40	2457.849	1253.396	2456.221	1289.169
50	2389.711	3019.254	2391.296	3312.419
60	2362.252	1258.342	2360.436	1002.875
70	2440.021	1718.656	2441.693	2057.711

Proto pro další experimenty již bylo využito nastavení počtu fuzzy systémů $N_f = 200$ a počtu fuzzy pravidel $N_r = 60$. Dalšími parametry, které mohou ovlivnit přesnost predikce je velikost populace P_s a pravděpodobnost mutace P_m . Tabulka 4 ukazuje výsledky experimentů pro $P_s = \{100, 150, 200\}$ a $P_m = \{0.01, 0.05\}$. Minimální chyby na testovacích datech bylo dosaženo pro $P_s = 200$ a $P_m = 0.05$.

Tabulka 4 - Výsledky genetického algoritmu SEFC pro dané nastavení parametrů. Zdroj: [vlastní]

P_s	P_m	Trénovací data		Testovací data	
		MSE	STD	MSE	STD
100	0.01	2383.813	2204.187	2383.654	2193.136
100	0.05	2360.647	1224.972	2359.429	1129.276
200	0.05	2264.103	407.0159	2264.475	793.574
150	0.05	2393.813	2314.187	2398.654	2289.136
150	0.01	2460.647	1115.972	2451.429	1021.276
200	0.01	2364.103	1007.015	2364.475	1293.5746

Při výběru optimálních hodnot parametrů navrženého modelu je nutností sledovat ukazatele kvality predikce. Logicky lze odvodit, že za nejlépe navržený model lze považovat takový model, jehož MSE a STD dosahuje nejnižší hodnoty. Při pohledu na tabulku 4 je patrné, že nejhodnější návrh parametrů modelu představuje nastavení parametrů SEFC, které je uvedeno v tabulce 5.

Tabulka 5 - Optimální parametry SEFC. Zdroj: [vlastní]

N_f	N_r	P_s	P_m	Trénovací data		Testovací data	
				MSE	STD	MSE	STD
200	60	150	0.05	2264.103	407.015	2264.475	793.574

Tabulka 5 prezentuje výsledek nastavení parametrů modelu predikce akciového kurzu pomocí algoritmu SEFC. Při nastavení parametrů $N_f = 200$, $N_r = 60$, $P_s = 150$ a $P_m = 0.05$ a vstupních normalizovaných datech, bylo dosaženo úspěšnosti predikce testovacích dat s průměrnou MSE = 2264.475 a průměrnou STD = 793.574. Změna konstanty K ovlivňující počet parametrů v pravidlech neměla na výsledky vliv.

5.3 Porovnání s výsledky metod učení s učitelem

Při učení s učitelem existuje nějaké vnější kritérium určující, který výstup je správný a v síti se nastavují váhy pomocí zpětné vazby podle toho, jak blízko je výstup kritériu. Vypočítává se rozdíl mezi žádaným a skutečným výstupem. Váhy se modifikují podle nějakého algoritmu, který zabezpečuje snižování chyby mezi skutečným a žádaným výstupem. Velikost změn vah v jednom kroku učení je obvykle malá. Pak se neuronové síti předloží nový vstup a celý proces se zopakuje. Po provedení velkého počtu pokusů se neuronová síť naučí vydávat stabilní výstup jako reakci na vstupy, které přijímá.

Při učení s učitelem se umělá neuronová síť učí srovnáváním aktuálního výstupu s výstupem požadovaným a přestavováním synaptických vah tak, aby se pro příště snížil rozdíl mezi skutečným a žádaným výstupem. Metodika snižování rozdílu je určena učícím algoritmem. Mezi metody učení s učitelem patří například stochastické učení a zpětné šíření chyby (Back-propagation). Pro porovnání výsledků byly použity metody učení s učitelem, dopředná neuronová síť MLP-BP a neuronová síť typu RBF.

5.3.1 Neuronová síť RBF

Struktura neuronové sítě radiálních bazických funkcí (RBF) se skládá z jedné vstupní, jedné skryté a jedné výstupní vrstvy. Tato skrytá vrstva obsahuje neurony, kde je každý z nich aktivován, jakmile vstup do sítě spadá blízko k bodu, který je považován za centrum tohoto neuronu. Konečný výsledek této sítě je poskytován neurony výstupní vrstvy, které vykonávají vážený součet pomocí výstupů ze skrytých neuronů. Výstup sítě je závislý na hodnotách

plynoucích z každého skrytého neuronu, stejně jako na vahách spojení mezi každým skrytým neuronem a každým výstupním neuronem. Výstup každého skrytého neuronu je výstupem z jeho aktivační funkce, a tato funkce je závislá na centru neuronu a parametru nazývaném poloměr. To znamená, že hodnota poskytovaná neuronem se mění v závislosti na vzdálenosti od vstupu do systému k centru neuronu, a podle hodnoty poloměru. Tímto způsobem mohou funkce získat svého maxima, když vstup je roven centru, ale další funkce mohou dosáhnout minima ve stejném bodě. Ačkoli lze jako aktivační funkce využít mnoho funkcí, pro skryté neurony se nejčastěji používá Gaussova funkce. Za účelem návrhu struktury neuronové sítě RBF je třeba zadat, kolik neuronů bude tvořit skrytou vrstvu. Poté se vybere počet bodů z trénovací množiny prostřednictvím K-Means shlukovacího algoritmu jako centra neuronů a stanoví se jednotný poloměr pro všechny neurony jako polovina průměrné vzdálenosti mezi množinou a centry. Jakmile se ustálí centra a poloměry neuronové sítě, množina vah může být vypočítána pomocí jakékoli metody vhodné k nalezení hodnoty, které minimalizují MSE. Proto je poloměr a centrum každého neuronu upraveno metodou delta pravidla. Parametrem této neuronové sítě je celé číslo označující počet neuronů ve skryté vrstvě. Toto číslo se mění v závislosti na řešeném problému. Výchozí hodnota expertně nastavena na 50 neuronů se ukázala jako příliš vysoká. Síť se učí velmi rychle a je vhodná pro regresi i klasifikaci.

Tabulka 6 - Výsledky predikce sítě RBF pro různý počet neuronů ve skryté vrstvě. Zdroj: [vlastní]

Počet neuronů ve skryté vrstvě	Trénovací data		Testovací data	
	MSE	STD	MSE	STD
10	39324.729	7348.211	39359.710	6569.587
15	14778.379	2057.137	14968.837	3215.122
20	13014.542	1302.665	12929.138	2005.720
25	42865.591	45960.946	42968.991	47000.755
30	18342.334	9077.894	18550.641	10211.144
35	30954.563	10624.740	31094.832	11966.790
40	162037.558	74459.706	161978.074	74549.412
45	745917.096	102473.625	750153.586	102648.746

Nepřesnějších výsledků predikce dosáhla síť RBF při nastavení počtu neuronů ve skryté vrstvě na 20 s průměrnou střední kvadratickou chybou $MSE = 12929.138$ a směrodatnou odchylkou $STD = 2005.720$ pro testovací data.

5.3.2 Dopředná neuronová síť typu perceptron

Třída dopředných neuronových sítí typu perceptron (MLP) se skládá z více vrstev neuronů, mezi sebou propojených dopředným způsobem. Každý neuron v jedné vrstvě má přímé spojení s neurony ve vrstvě následující. V mnoha aplikacích jednotky těchto sítí používají sigmoidální¹² aktivační funkce. Jako algoritmus učení se používá metoda zpětného šíření chyby (Backpropagation). Výstupní hodnoty jsou zde porovnávány se správnou odpovědí pro výpočet hodnoty některé předdefinované chybové funkce. Pomocí této informace algoritmus upravuje váhy jednotlivých spojení s cílem snížit hodnotu chybové funkce alespoň o nějaké malé množství. Po opakování tohoto procesu pro dostatečně velký počet trénovacích cyklů, MLP obvykle dokonverguje do nějakého stavu, kde je chyba výpočtů malá. MLP je nejrozšířenější a nejpoužívanější síť s učením s učitelem. Používá se jak pro predikci časových řad, regresi tak i pro klasifikaci. K nevýhodám sítě patří obtížné řešení problému lokálních minim a poměrně dlouhá doba učení. Mezi důležité parametry sítě patří počet skrytých vrstev, počet neuronů v každé skryté vrstvě, typ aktivační funkce neuronu a rychlost učení. Tabulka 7 ukazuje výsledky experimentů pro MLP, kde byl nastavován počet skrytých vrstev 1 nebo 2, měněn počet neuronů ve skryté vrstvě, aktivační funkce (logistická nebo hyperbolický tangens) a rychlost učení (0.05, 0.1 nebo 0.15).

Neuronová síť MLP dosáhla nejlepších výsledků predikce při nastavení 2 skrytých vrstev, 5 neuronů v každé skryté vrstvě, logistické aktivační funkce neuronů a rychlosti učení 0.15. Průměrná střední kvadratická chyba testovacích dat byla $MSE = 6170.371$ a průměrná směrodatná odchylka $STD = 2923.782$.

¹² Sigmoidální přenosová funkce je ve tvaru $f(x) = \frac{1}{1+e^{-kx}}$. Její hodnoty se blíží v mínus nekonečnu nule a v plus nekonečnu jedné. Pro nulu je hodnota 0.5. Výhodou sigmoidální přenosové funkce oproti skokové je existence spojité první derivace v každém bodě.

Tabulka 7 - Výsledky predikce MLP pro dané nastavení parametrů. Zdroj: [vlastní]

Počet skrytých vrstev	Počet neuronů ve skryté vrstvě	Aktivační funkce neuronu	Rychlost učení	Trénovací data		Testovací data	
				MSE	STD	MSE	STD
2	3	log	0.15	6353.159	689.748	6967.112	5035.267
2	15	log	0.15	5046.761	889.924	6207.707	3325.157
2	10	log	0.15	5542.600	1482.190	6816.795	3099.977
2	5	log	0.15	4962.816	402.808	6170.371	2923.782
1	10	log	0.1	6649.033	346.532	8244.881	3511.600
1	5	log	0.05	6767.054	330.915	7763.427	3648.754
1	15	log	0.05	6818.599	227.280	7852.388	2843.484
2	15	htan	0.15	23665.156	6065.052	21933.565	4517.945
1	10	htan	0.1	21524.337	2658.938	24595.637	7388.839

5.3.3 Porovnání výsledků predikce

Z uvedené tabulky je patrné, že nejlepších výsledků predikce dosáhl genetický algoritmus posilovaného učení SEFC se střední kvadratickou chybou odhadu MSE = 2264.475 pro testovací data a MSE = 2264.103 pro data trénovací.

Z metod učení s učitelem dosáhla lepších výsledků neuronová síť MLP před sítí RBF, kde střední kvadratická chyba dosáhla hodnoty MSE = 6170.371 pro testovací data a hodnoty MSE = 4962.816 pro trénovací data. Použití sítě RBF již nedosahuje tak přesných výsledků jako předchozí metody, viz tabulka 8. Výsledky jsou uvedeny pro jednotlivé složky dat pro $p = 10$, kde vždy 9 podmnožin A slouží jako trénovací a 1 jako testovací data.

Tabulka 8 - Porovnání výsledků metod. Zdroj: [vlastní]

Datové složky	SEFC		RBF		MLP	
	MSE testovací	MSE trénovací	MSE testovací	MSE trénovací	MSE testovací	MSE trénovací
1	2292.233	2342.076	17848.672	12852.891	5585.680	4778.385
2	2293.998	2210.908	12143.963	12875.865	3541.842	4789.291
3	2340.456	2302.361	12608.214	13022.008	6001.498	5533.097
4	2052.980	2243.121	14699.719	13011.252	5742.428	4299.837
5	2299.502	2301.721	13423.992	12962.080	4846.368	5474.984
6	2234.284	2266.090	11169.356	13191.816	5631.148	4998.963
7	2223.635	2247.434	12152.566	13210.377	4943.047	5139.358
8	2291.803	2227.313	10156.986	13187.987	14715.700	4319.960
9	2280.134	2218.986	12893.273	12884.222	5127.717	5241.082
10	2340.721	2288.017	12198.630	12948.912	5568.281	5053.207
Celková MSE	2264.475	2264.103	12929.137	13014.541	6170.371	4962.816
Celková STD	793.574	407.015	2005.719	1302.665	2923.782	402.808

6 Uživatelská příručka Keel 2.0

Keel [16] je open source (GPLv3) Java softwarový nástroj pro posouzení evolučních algoritmů pro problémy data miningu, včetně regrese, klasifikace, shlukování, pattern mining¹³ atd. Obsahuje velkou sbírku klasických algoritmů extrakce znalostí, techniky předzpracování (výběr tréninkové množiny, výběr příznaků, diskretizace, dopočtové metody pro chybějící hodnoty, atd.) Algoritmy učení založené na výpočetní inteligenci, včetně evolučních pravidlových algoritmů učení založených na různých přístupech (Pittsburgh, Michigan a IRL atd.) a hybridních modelů, jako jsou genetické fuzzy systémy, evoluční neuronové sítě, atd. To umožňuje provést úplnou analýzu všech modelů učení ve srovnání s existujícím, včetně statistických testovacích modulů pro srovnání. Kromě toho byl Keel navržen s dvojitým cílem, a tím je výzkum a vzdělávání (obr. 10).



Obrázek 10 - Softwarový nástroj Keel. Zdroj: [vlastní]

Keel je softwarový nástroj, pro vytváření a používání různých Data miningových modelů a je prvním softwarovým nástrojem tohoto typu obsahujícím knihovnu volných kódů v jazyce Java pro evoluční algoritmy učení. Hlavními rysy Keelu jsou následující [16]:

- Obsahuje algoritmy pro předzpracování: transformace, diskretizace, výběry instancí a selekce významných atributů.
- Obsahuje také knihovnu algoritmů extrakce znalostí, učení s učitelem a bez učitele, poznamenávající začlenění mnohočetných evolučních algoritmů učení.
- Součástí je knihovna statistických analýz pro analýzu algoritmů.

¹³ Pattern mining je metoda dolování dat, která zahrnuje hledání existujících vzorů v datech. V této souvislosti se vzory často rozumí asociační pravidla. Původní motivace pro hledání asociačních pravidel pochází z přání analyzovat transakční data supermarketu, tedy zkoumat chování zákazníků v z hlediska zakoupených produktů.

- Obsahuje uživatelsky přívětivé rozhraní, zaměřené na analýzu algoritmů.
- Prostředí Keelu lze připojit k internetu ke stažení nové datové soubory pro jejich použití v dalších analýzách.

6.1 Příprava dat

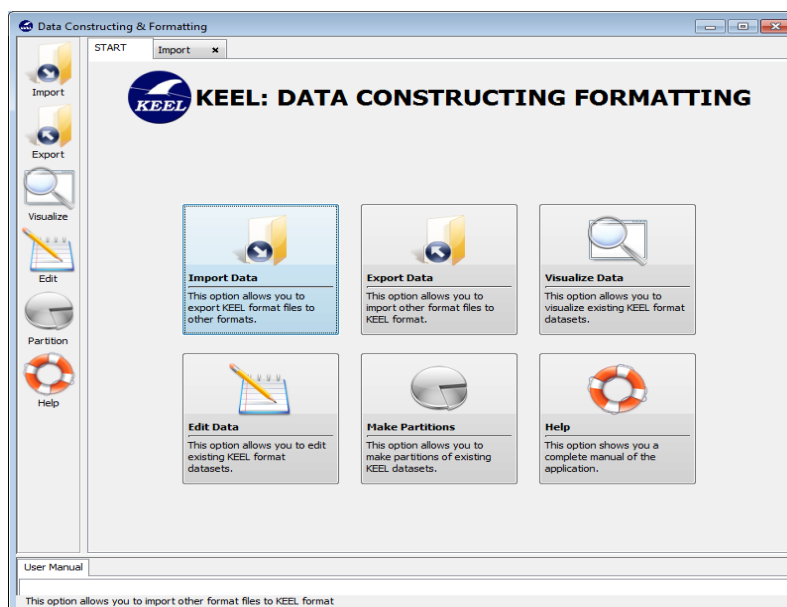
Vizualizace dat: Tato volba umožňuje zobrazit stávající datové soubory ve formátu Keel.

Import dat: Tato volba umožňuje exportovat soubory ve formátu Keel do jiných formátů.

Export dat: Tato volba umožňuje importovat soubory jiného formátu do Keel formátu.

Tvorba oddílů: Tato volba umožňuje vytvořit oddíly existujících souborů dat.

Editace dat: Tato funkce umožňuje editovat existující soubory dat ve formátu Keel (obr. 11).

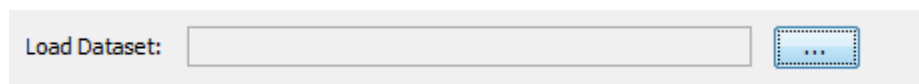


Obrázek 11 - Datový management v Keel. Zdroj: [vlastní]

Existují různé formáty dat, se kterými lze pracovat pomocí nástroje Keel. CSV formát je implementací souboru s vymezeným textem, který používá k oddělení hodnot čárku. CSV formát je velmi jednoduchý a podporovaný téměř všemi tabulkovými a databázovými systémy řízení. Text oddělený tabulátory (TXT) nebo hodnoty oddělené tabulátorem (TSV) jsou jednoduchá textová data umožňující výměnu tabulkových dat mezi aplikacemi s různým vnitřním formátem. Hodnoty oddělené tabulátory byly úředně registrovány, jako MIME typ. PRN formát má stejné vlastnosti a omezení jako formát CSV, rozdíl je v oddělovačích mezi poli, kterými jsou ve formátu PRN mezery. Nicméně, mezery mají v tomto formátu odlišnou funkci, než v CSV souborech. DIF je textový formát, který se používá pro import a export mezi různými tabulkovými programy jako je Excel, StarCalc, dBase atd. Tento typ formátu je uložen s příponou dif. Soubory kódované podle C4.5 formátu se skládají

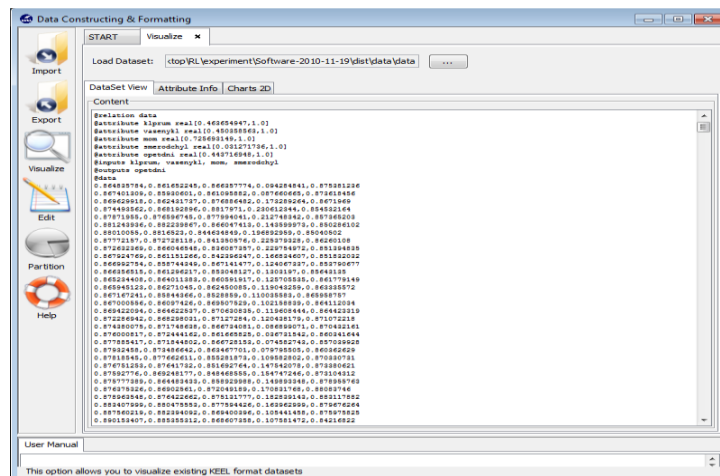
ze dvou souborů, jeden z nich je jmenný soubor s příponou ". jména", druhý, že je datový soubor s příponou ". data". Datový formát excelu, tabulkového programu napsaného a distribuovaného společností Microsoft. V současné době je nejpoužívanějším tabulkovým programem pro operační systémy Microsoft Windows a Apple Macintosh. Tabulkový procesor je program, který umožňuje manipulaci s číselnými a alfanumerickými daty. Tabulkové procesory jsou uspořádány v řádcích a sloupcích. Průsečík řádku a sloupce se nazývá buňka. Každá buňka může obsahovat data nebo vzorce, které mohou odkazovat na obsah ostatních buněk atd. Mezi další podporované datové formáty patří HTML, XML a WEKA.

Existují různé možnosti, jak zobrazovat informace, lze zobrazit obsah dat, specifické informace o attributech a porovnat dva atributy pomocí grafu. Nejprve je třeba vybrat cestu zdrojových dat (v Keel formátu), které se mají zobrazit (obr. 12). Po načtení souboru, v závislosti na zvolené možnosti jsou uvedeny různé informace o souboru dat.



Obrázek 12 - Načtení datového souboru. Zdroj: [vlastní]

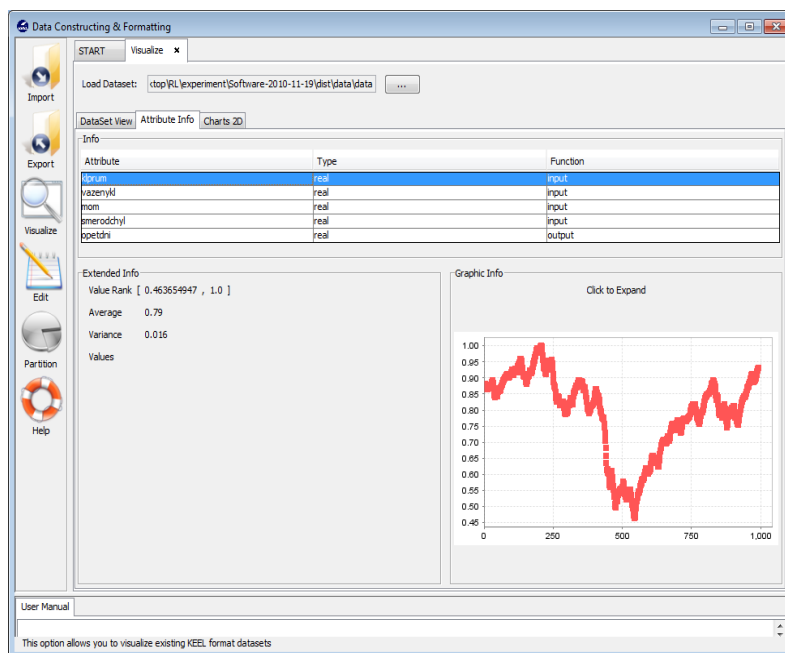
Vyberete-li možnost zobrazení datové sady, lze zobrazit obsah vybraného souboru dat. Tyto informace však nelze změnit, je možné je pouze vizualizovat (obr. 13).



Obrázek 13 - Vizualizace dat. Zdroj: [vlastní]

Informace o attributech. V této možnosti lze získat podrobné informace o attributech definovaných v souboru dat. Informace v horní části okna jsou: typ atributu (integer, real nebo nominal) a zda je atribut vstupní nebo výstupní. Bližší informace lze vidět v levé dolní části okna. Zobrazená informace závisí na typu atributu, v případě integer nebo real atributu je uveden rozsah hodnot, průměr a rozptyl. V případě nominálního atributu, lze

zobrazit pouze jeho možné hodnoty. Konečně v pravé dolní části okna je zobrazen graf s rozložením hodnot atributu (obr. 14).

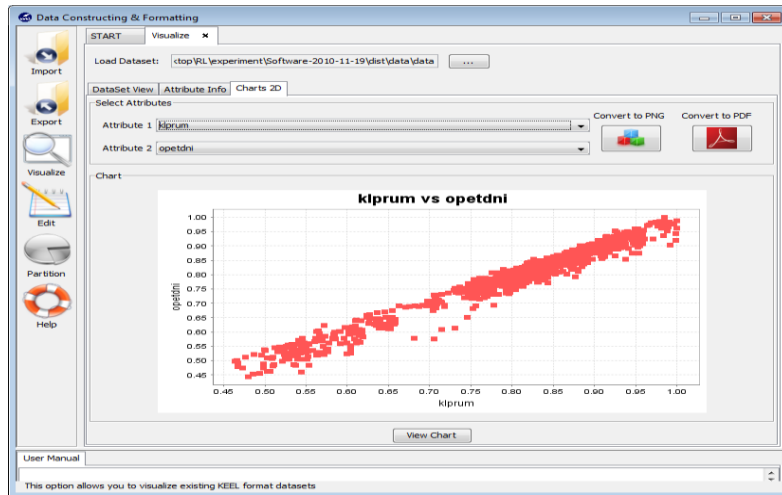


Obrázek 14 - Záložka informací o attributech. Zdroj: [vlastní]

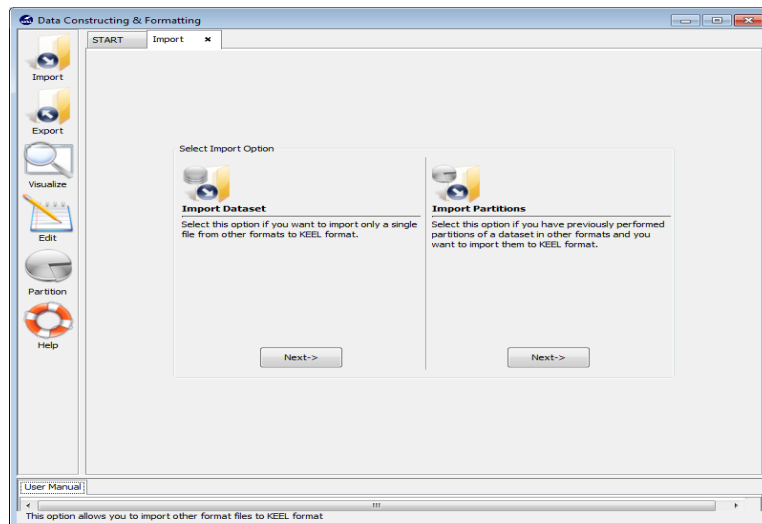
Ve volbě graf 2D lze porovnat různé atributy (obr. 15). Nejprve je nutné vybrat dva atributy, které chcete porovnat. Pro každý atribut je k dispozici seznam všech atributů souboru dat. Jakmile jsou vybrány atributy, kliknutím na "zobrazit graf" se grafika zobrazí. Pokud potřebujete vložit vytvořený graf do jiného dokumentu, lze použít tlačítka "převést do PNG" a tato možnost uloží graf jako obrázek ve formátu PNG. "Převod do formátu PDF" uloží graf jako PDF dokument.

Import dat umožňuje převádět soubory v různých formátech (TXT, Excel, XML, atd.) do formátu Keel. Následující obrázek ukazuje dvě možnosti jak importovat data. První z nich je import jedné datové sady, další možností je import oddílů souboru, které jsou k dispozici v jiných formátech než formát Keel. Dále je popsán proces možností importování (obr. 16).

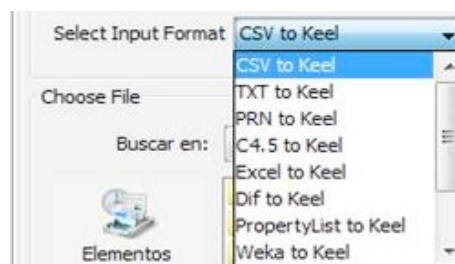
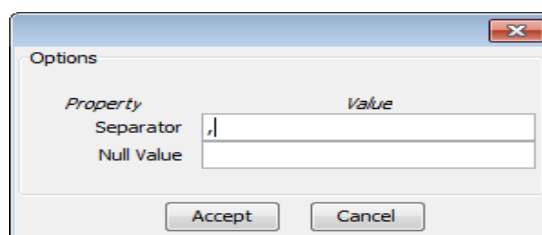
Import datové sady umožňuje importovat pouze jeden soubor z jiných formátů do formátu Keel. Nejprve je nutné vybrat zdrojový formát souboru datové sady. Mezi povolenými formáty jsou CVS, TXT, PRN, C4.5, Excel, Dif, PropertyList a Weka. Tlačítkem "Možnosti" lze nastavit separátor v případě, že je nutný a zda je použita nulová hodnota ve zdrojovém souboru (obr. 17). Jakmile byl typ konverze a zdrojový soubor nakonfigurován, je třeba kliknout na tlačítko Další, poté je zobrazen originální a importovaný soubor.



Obrázek 15 - Zložka porovnání atributů. Zdroj: [vlastní]

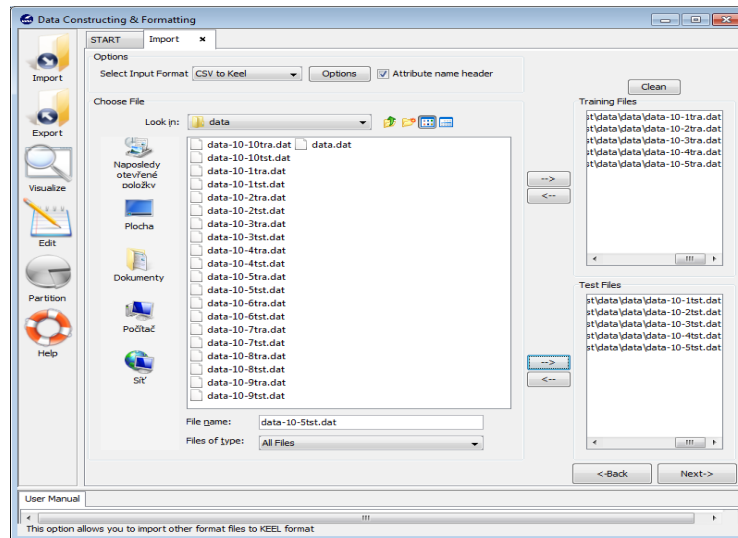


Obrázek 16 - Importování datové sady a oddílů. Zdroj: [vlastní]



Obrázek 17 - Výběr vstupního formátu, nastavení separátoru a nulové hodnoty. Zdroj: [vlastní]

Potvrzením konverze existují dvě možnosti uložení importovaného souboru. Pokud je označena možnost import do sekce experimentů, importovaný datový soubor bude zahrnut jako možnost v Keel experimentech. Tato datová sada tak bude k dispozici ke zpracování pomocí Keel metod. Pokud si nepřejete tuto možnost, kliknutím na tlačítko Uložit, je zvolen cílový adresář pro transformovaný datový soubor (obr. 18).



Obrázek 18 - Import trénovacích a testovacích souborů. Zdroj: [vlastní]

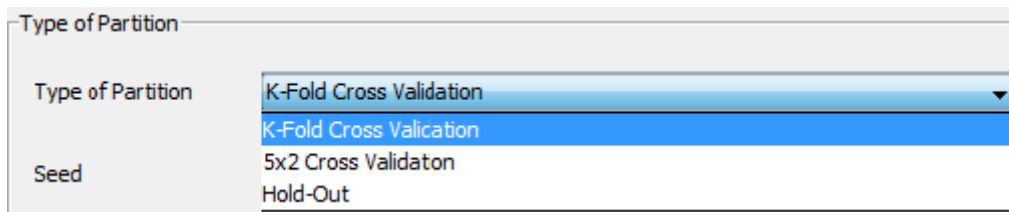
Import oddílů lze vybrat, pokud je již dříve provedeno rozdělení souboru dat do jiných formátů a chcete ho importovat do Keel formátu. Tato volba umožňuje samostatně zvolit trénovací a testovací soubory. Po zvolení zdrojového formátu souboru, je možné nastavit separátor či použití nulových hodnot. Další postup je stejný jako při importu datové sady.

Stejně jako při importování dat, zde existují dvě možnosti exportu tj.: import jedné datové sady a export oddílů souboru. Oba typy exportu se skládají ze tří kroků, kterými jsou výběr zdrojového souboru, volba vstupního formátu a uložení souboru.

Rozdělení dat umožňuje vytvořit oddíly z existujícího datového souboru v Keel. Tento proces se skládá z následujících kroků (obr. 19):

- 1) Výběr souboru - nejprve je nutné zvolit cestu pro kompletní datovou sadu, z které se mají vytvořit oddíly.
- 2) Typ rozdělení - jakmile je soubor vybrán, je nutné zvolit typ rozdělení.

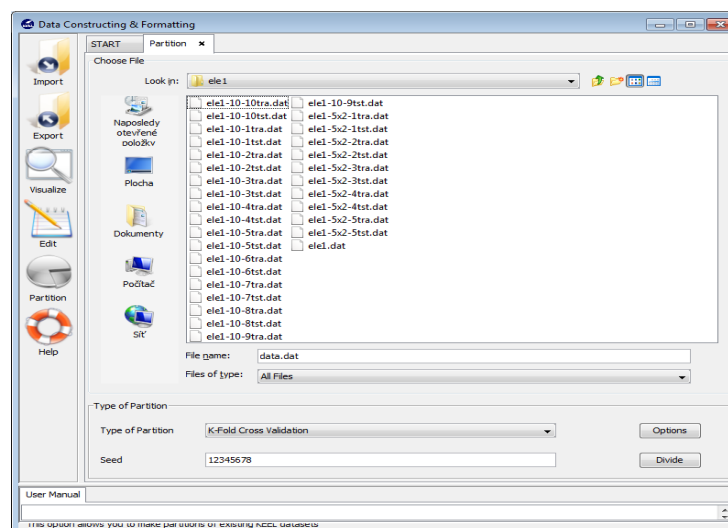
- K-násobná křížová validace - toto rozdělení umožňuje nastavit počet složek, do kterých se má soubor rozdělit. Výchozí hodnota je nastavena na 10 (nastavení jiného počtu lze provést kliknutím na tlačítko "Možnosti").
- 5 × 2 křížová validace – toto rozdělení neumožňuje změnu nastavení
- Hold-out - toto rozdělení umožňuje nastavit počet oddílů a procentní podíl trénovací a testovací množiny



Obrázek 19 - Výběr typu rozdělení dat. Zdroj: [vlastní]

3) Seed – k vytvoření oddílů datového souboru je nutné specifikovat generátor náhodných čísel k provedení procesu rozdělení.

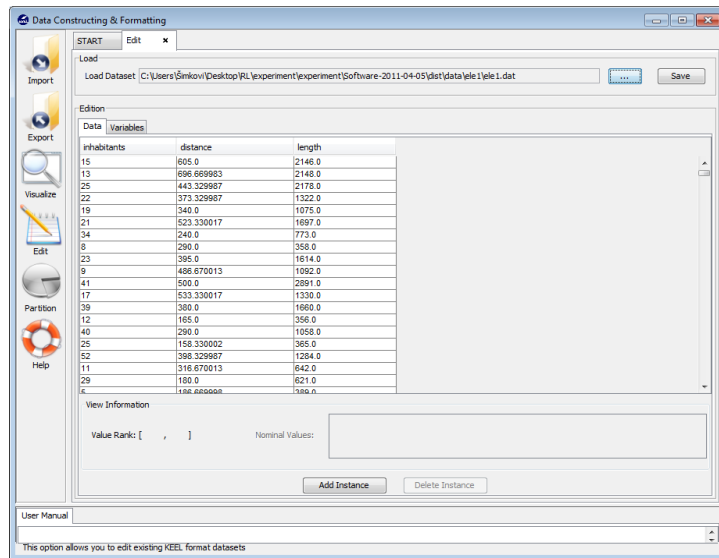
4) Vytvoření oddílů - jakmile byly předchozí kroky nastaveny, rozdělení bude provedeno kliknutím na tlačítko "Divide". Soubory vytvořené v průběhu procesu rozdělení budou uloženy v adresáři zadaném v kroku 1 a název těchto souborů je uveden automaticky a závisí na typu vybraného rozdělení (obr. 20).



Obrázek 20 - Vytvoření jednotlivých oddílů dat. Zdroj: [vlastní]

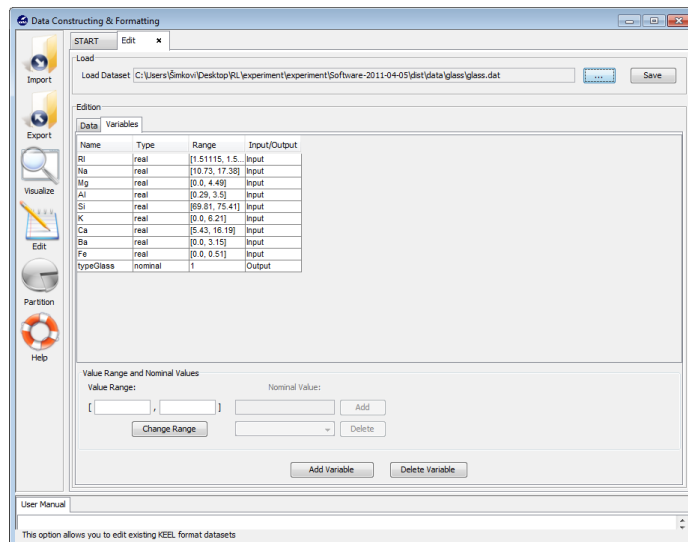
Editace dat umožňuje upravovat existující data ve formátu Keel za účelem přidání nových atributů, odstranění atributů, opravení některých chyb atd. Nejprve je nutné nastavit cestu pro zdrojová data. Úpravy mohou být provedeny v instancích i v proměnných.

1) Editace dat - V případě této možnosti mohou být přidány nové instance nebo existující instance lze odstranit či upravit. Lze tedy provádět následující operace: modifikace obsahu buňky, smazání a přidání nové instance (obr. 21).



Obrázek 21 - Editace dat. Zdroj: [vlastní]

2) Editace proměnných - V této volbě je možné provádět různé modifikace proměnných. Pomocí tabulky a tlačítek umístěných nad nimi lze provést přidání nových proměnných, změnit rozsah hodnot proměnné, změnit nominální hodnoty, změnit typ a dále funkci atributu, zda se jedná o vstupní či výstupní atribut. Jakmile jsou dokončeny všechny změny, lze je uložit do souboru stisknutím tlačítka pro uložení (obr. 22).

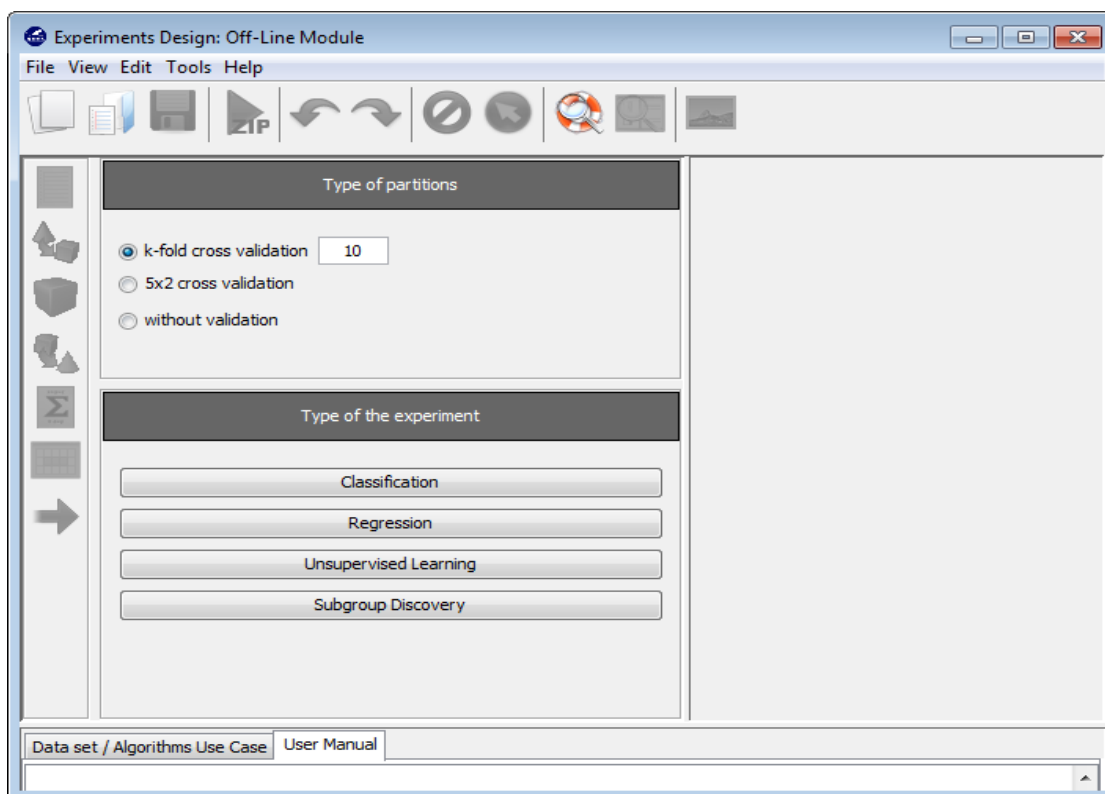


Obrázek 22 - Editace proměnných. Zdroj: [vlastní]

6.2 Návrh experimentu

Návrh experimentů má za cíl navrhnout požadované experimenty s použitím grafického rozhraní. Bezpochyby se jedná o nejvíce inovativní nástroj integrovaný v tomto programu. Cílem je používat dostupné datové soubory a algoritmy pro generování adresářové struktury se všemi potřebnými soubory nutnými ke spuštění navrženého experimentu vybraného uživatelem v místním počítači. S tímto programem, stačí pouze zvolit vstupní data (datové soubory), algoritmy, které chcete použít a učinit vhodné propojení mezi nimi. Také je možné metody spojovat, vkládat statistické testy atd. Úkol, který je zjednodušený, je pravděpodobně nastavení parametrů, vše je možné provádět z jednoduchého dialogu bez požadavku externích konfiguračních souborů. Tato část Keelu má dva hlavní cíle: na jedné straně, lze použít software pro testovací a hodnotící nástroj při vývoji algoritmů. Na druhé straně, je také dobrou volbou pro porovnání nově vyvinutých a standardních algoritmů již zavedených a dostupné v Keel 1.0.

Je-li zvolena možnost experimentování, objeví se hlavní okno modulu experimentů (obr. 23).



Obrázek 23 - Hlavní okno modulu experimentů. Zdroj: [vlastní]

Nejprve je nutné zvolit typ experimentu a typ rozdělení dat. Vybrané možnosti určí druh metod a dat, které jsou dostupné pro provedení experimentu.

Jsou dostupné následující typy rozdělení dat:

- k-násobná křížová validace,
- 5×2 křížová validace,
- žádná validace.

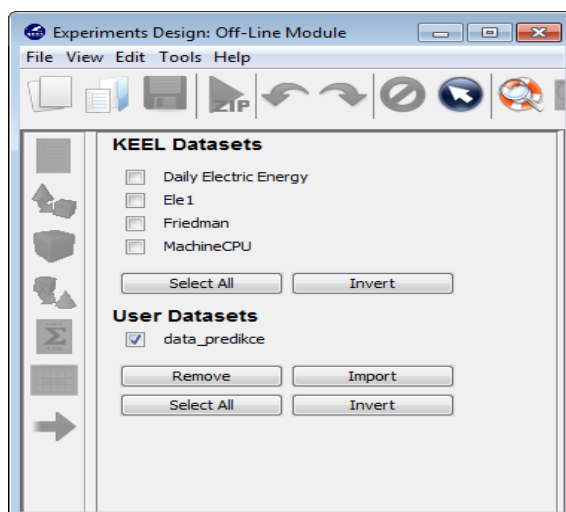
Modul experimentů v aktuální verzi Keelu nabízí následující typy experimentů:

- klasifikace,
- regrese,
- učení bez učitele.

Jakmile je vybrán typ experimentu, zobrazí se panel výběru datových souborů, který umožní pokračovat v návrhu experimentu.

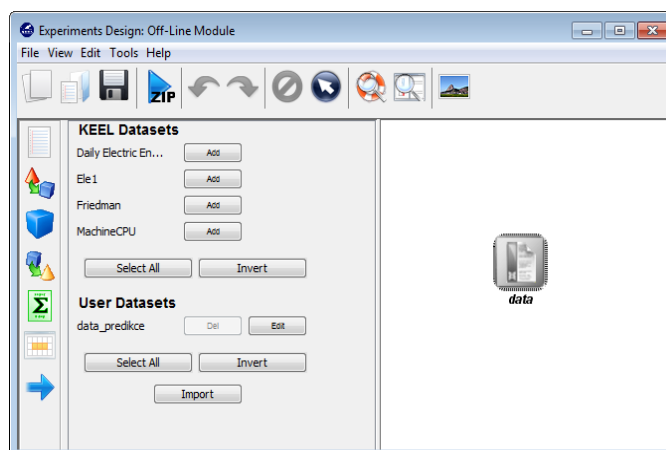
Panel výběru datových souborů zobrazuje dostupné datové soubory pro aktuální experiment (obr. 24). Jeho obsah bude záviset na vybraném typu experimentu. Dalším krokem je vybrání zvolených dat z panelu. Tlačítka vybrat vše a invertovat umožňují zjednodušení výběru.

Tlačítko Import umožňuje importovat existující datový soubor do prostředí Keelu pro aktuální experiment. Po kliknutí se zobrazí hlavní okno nástroje pro importu dat. Tento proces je popsán v kapitole příprava dat.



Obrázek 24 - Výběr datových souborů pro experiment. Zdroj: [vlastní]

Jakmile jsou vybrána všechna potřebná data, lze pokračovat v procesu návrhu experimentu. Pro nastavení uzlu datových souborů experimentu, musí uživatel kliknout na bílý panel grafu (obr. 25). Pokud jsou nalezeny nějaké chybějící oddíly, nástroj okamžitě zobrazí zprávu, zda se mají automaticky vygenerovat chybějící oddíly.



Obrázek 25 - Vložení datového uzlu. Zdroj: [vlastní]

6.3 Složky panelu experimentu

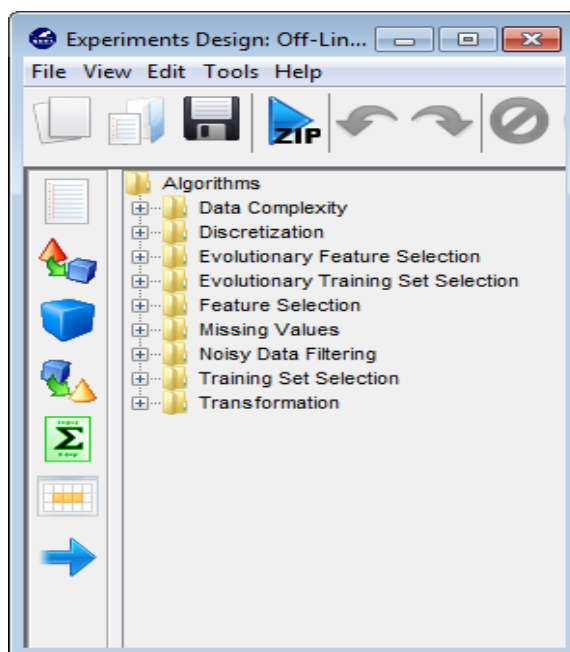
Panel experimentu zobrazuje složky stávajícího experimentu a popisuje vztahy mezi nimi. Uživatel může přidávat nové složky pomocí levého menu. V tomto menu jsou k dispozici následující kategorie:



Datové soubory – tento modul umožňuje uživateli editovat aktuální vybraná data pro experiment. Uživatel může stále ještě přidat nebo odstranit data pro experimentu a také je možné importovat nové datové soubory. Navíc, tlačítko edit umožňuje uživateli určit, které oddíly (trénovací a testovací) mají být použité. Tímto způsobem je možné dočasně měnit soubory, které budou zahrnuty do experimentu.



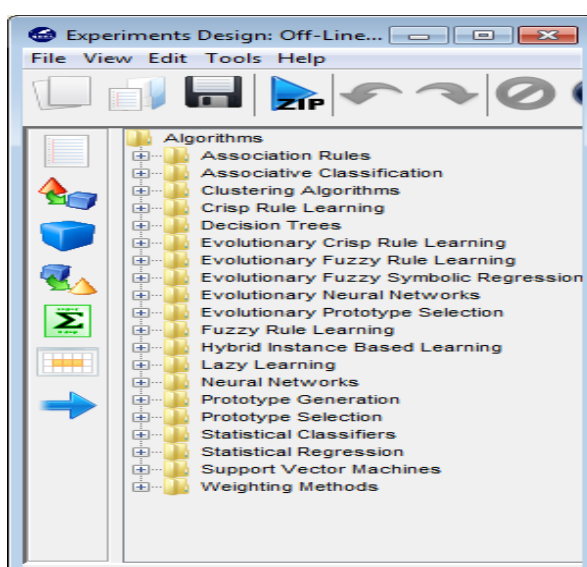
Metody předzpracování – tato kategorie zahrnuje několik metod pro předzpracování dat (obr. 26), např.: metody pro transformaci formátu dat z nominálních do binárních, desetinné škálování; metody pro převod reálných nebo číselných dat do nominálních dat; metody výběru vlastností dat; metody výběru instancí v datech a generování nových instancí z originálních instancí; metody pro posouzení dat obsahující chybějící hodnoty atd.



Obrázek 26 - Přehled metod pro předzpracování. Zdroj: [vlastní]



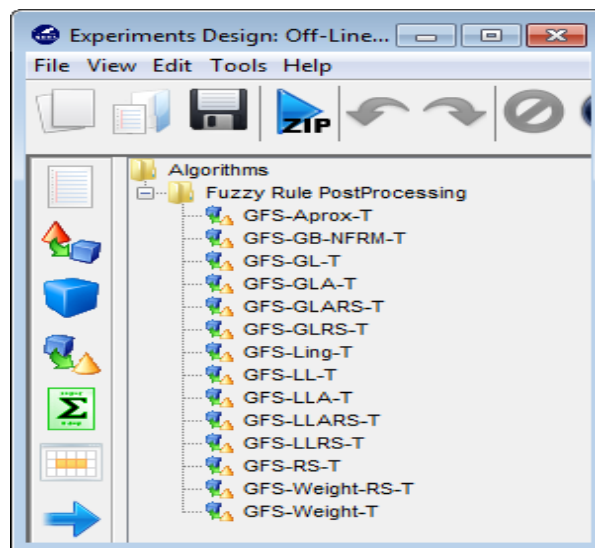
Standardní metody – kategorie obsahuje data miningové metody obsažené v nástroji Keel jako jsou např.: metody pro extrahování asociačních pravidel z dat, shlukovací metody, metody pro tvorbu rozhodovacích stromů, umělé neuronové sítě, klasifikátory založené na statistických modelech, metody učení založené na pravidlech a fuzzy pravidlech, konstrukce statistických modelů, support vector machine, metody symbolických regresních procedur, evoluční učení založené na fuzzy pravidlech, metody zjišťování podskupin a mnoho dalších (obr. 27).



Obrázek 27 - Přehled dostupných metod a algoritmů. Zdroj: [vlastní]



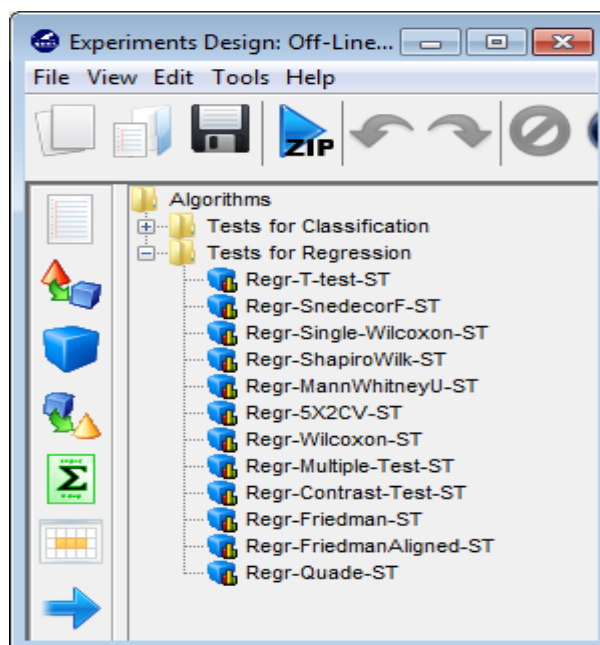
Metody následného zpracování – obsahují metody následného zpracování výsledků pomocí fuzzy pravidel (obr. 28).



Obrázek 28 - Metody pro následné zpracování výsledků. Zdroj: [vlastní]



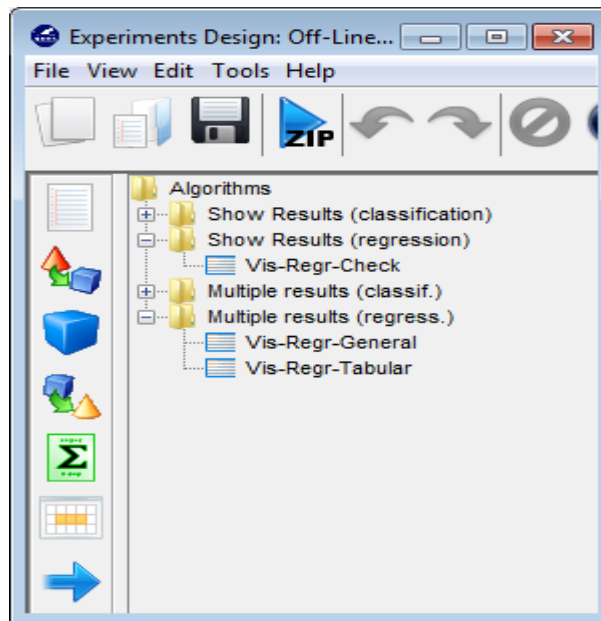
Statistické testy – obsahují statistické procedury pro porovnání výsledků regresních a klasifikačních experimentů (obr. 29).



Obrázek 29 - Statistické testy pro regresi a klasifikaci. Zdroj: [vlastní]



Vizualizační moduly – tato kategorie obsahuje několik vizualizačních modulů pro analýzu a sumarizaci výsledků klasifikace a regrese dosažených v experimentech (obr. 30).

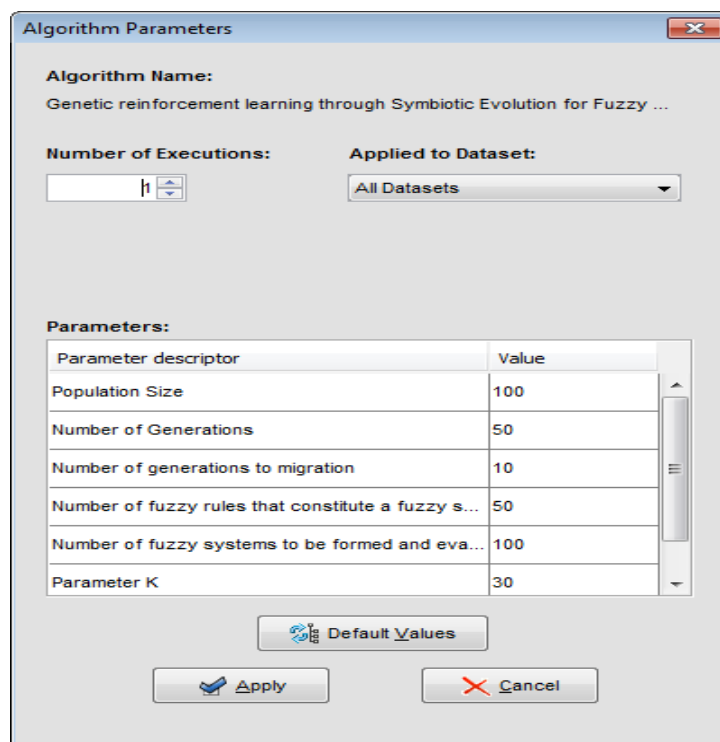


Obrázek 30 - Vizualizační moduly pro analýzu a sumarizaci výsledků. Zdroj: [vlastní]



Propojování - připojení umožňují dokončení návrhu experimentu, spojením zahrnutých modulů s toky, které představují tok dat v experimentu. Mohou být použity jako vstupy a výstupní moduly. Existují zde určitá omezení, která je třeba vzít v úvahu při propojování mezi různými prvky jako např.: datový soubor nemá vstupy, vstupem pro algoritmy předzpracování mohou být pouze datové soubory nebo jiné metody předzpracování, testovací a vizualizační moduly nemají výstupy, jako vstupy mohou být použita data získaná z metod nebo algoritmů následného zpracování, graf nemůže mít uzavřený cyklus. Všechna tato omezení jsou ověřena při vytváření spojení. Pokud některé z těchto spojení není povoleno, aplikace zobrazí chybové hlášení.

Jakmile je vložen modul v grafu, je možné nastavit hodnoty jeho parametrů (obr. 31). Kliknutím pravým tlačítkem myši na symbol algoritmu se zobrazí následující dialogová tabulka. V horní části tohoto dialogu je možné nastavit, kolikrát bude daný algoritmus vykonán (k dispozici pouze pro náhodné metody). V tabulce ve střední části okna jsou všechny parametry algoritmu nastaveny na výchozí hodnoty. Tyto hodnoty mohou být upraveny, podle vhodnosti pro konkrétní metodu, jinak se zobrazí chybová zpráva.



Obrázek 31 - Nastavení parametrů algoritmu. Zdroj: [vlastní]

6.4 Spuštění experimentu

Jakmile byl experiment navržen, uživatel ho může vygenerovat přes možnost Spustit experiment v liště nástrojů. V tuto chvíli, softwarový nástroj provede několik testů týkajících se kompletnosti experimentu. Zjistí-li, že některé chybějící oddíly datových souborů, nebo že některé prvky nejsou propojeny, tak zde je poslední možnost jejich opravení, jinak bude experiment vygenerován nesprávně. Pokud je vše v pořádku, uživatel vybere cestu pro uložení zip souboru obsahující experiment. Proces generování vytváří zip soubor, který obsahuje všechny prvky potřebné pro spuštění experimentu. Pokud je proces generování experimentu úspěšně dokončen, zobrazí se následující zpráva (obr. 32).



Obrázek 32 - Zpráva o procesu generování experimentu. Zdroj: [vlastní]

Experiment je třeba spustit pomocí souboru RunKeel.jar umístěném ve složce "experiment / scripts". Spuštěním souboru jsou vytvořeny následující čtyři adresáře:

- Exe adresář - obsahuje všechny jar soubory algoritmů vložených do experimentu.
- Adresář skriptů - obsahuje konfigurační soubory seřazené podle algoritmu. Samozřejmě také obsahuje RunKeel.jar soubor, který se používá ke spuštění experimentu.
- Adresář datových souborů - obsahuje použité datové soubory v experimentu. Pro každý soubor dat je vytvořen adresář.
- Adresář výsledků - obsahuje výstupních soubory z jednotlivých algoritmů.

Závěr

První část této práce je zaměřena na všeobecný popis současného stavu predikce akciových kurzů, základní pojmy a jednotlivé akciové analýzy. Následně se podrobně věnuji charakteristice posilovaného učení a jeho metodám.

Posilované učení pracuje s průběžně se učícím agentem na základě interakcí s prostředím, které je nedostupné a nedeterministické, přičemž zpětná vazba (odměna nebo trest) přichází jen někdy a opožděně. Agentovi není řečeno, které akce přijmout, stejně jako ve většině forem strojového učení, ale místo toho musí zjistit, jaké akce přinášejí největší odměnu tím, že je vyzkouší. Ve většině zajímavých a náročných případů mohou akce ovlivnit nejenom bezprostřední odměnu, ale i další situace, prostřednictvím všech následujících odměn. Tyto dvě charakteristiky, hledání pokusem a omylem a zpožděné odměňování, jsou dva nejdůležitější rozlišovací znaky posilovaného učení. Posilované učení není definováno charakteristickými metodami učení, ale charakterizuje jej učební problém. Každá metoda, která je vhodná k řešení tohoto problému, je považována za metodu posilovaného učení. Základní myšlenkou je jednoduše zachytit nejdůležitější aspekty skutečného problému, kterému čelí učící se agent interagující s jeho prostředím za účelem dosažení cíle. Je jasné, že takový agent musí být schopen do určité míry vytušit stav prostředí a musí být schopen přijímat akce, které ovlivňují stav. Agent musí mít také cíl nebo cíle týkající se stavu prostředí. Formulace má obsahovat právě tyto tři aspekty - vnímání, akci, a cíl - v jejich co možná nejjednodušší podobě.

Posilované učení se liší od učení s učitelem, metodě učení zkoumaném ve většině aktuálních výzkumů ve strojovém učení, statistických rozpoznávání vzorů a umělých neuronových sítí. Učení s učitelem je učení se z příkladů poskytnutých znalými externími učiteli. Je to důležitý druh učení, ale sám o sobě není dostačující pro učení se vzájemným ovlivňováním. V problémech interakcí je často nepraktické získat příklady žádoucího chování, které jsou současně správné a reprezentují veškeré situace, v níž má agent jednat. V neprobádaných oblastech, kde by se dalo očekávat, že učení je nejpřínosnější, agent musí být schopen poučit se z vlastní zkušenosti.

Jedním z větších trendů v posilovaném učení je oblast, kde dochází k většímu spojení mezi umělou inteligencí a dalšími technickými disciplínami. Není to tak dávno, co umělá inteligence byla vnímána jako téměř úplně oddělená od teorie řízení a statistiky. Muselo se pracovat s logikou a symboly, nikoliv čísly. Během posledních desetiletí se tento pohled

postupně začal měnit. V posilovaném učení se rozšířily myšlenky z teorie řízení a stochastických aproximací k řešení širších a ambicióznějších cílů umělé inteligence.

V další části práce jsem provedl predikci akciových kurzů pomocí modelu, založeném na posilovaném učení, konkrétně genetickém algoritmu posilovaného učení. Vstupními daty pro predikci byla zvolena finanční časová řada obsahující hodnoty nejsledovanějšího akciového indexu Nasdaq Composite, měřícímu vývoj všech akcií obchodovaných na The Nasdaq Stock Market. Předzpracovaná data byla importována do programového prostředí Keel a následně realizován navržený model pro predikci. Pro porovnání výsledků byly také využity metody učení s učitelem a to neuronová síť RBF a dopředná neuronová síť MLP, které nedosáhly zdaleka tak dobrých výsledků predikce. Nejlepších výsledků dosáhl genetický algoritmus posilovaného učení s průměrnou střední kvadratickou odchylkou přibližně 2264 pro trénovací i testovací data a ukázal se tak jako vhodný pro predikci akciových kurzů.

Co se týká výsledků investování, na sledovaných datech bylo možné dosáhnout pomocí pasivní indexové strategie výnosu 9.04%. Pokud by investor realizoval aktivní investiční strategii založenou na predikci pomocí modelu posilovaného učení navrženého v této práci, dosáhl by za toto období (2007-2010) výnosu 49.58% na testovacích datech. Pokud by do výsledku investování byly zahrnuty transakční náklady ve výši 1% nákladů z výše transakce, byl by tento výnos 44.09%. Tato strategie je založena na předpokladu, že nákup je realizován při očekávaném růstu ve výši minimálně 1% a prodej při očekávaném poklesu ve výši minimálně 1%. Tyto výsledky bude potřeba ověřit na delší časové řadě vzhledem k tomu, že v tomto období došlo vlivem finanční krize k výraznému poklesu akciového indexu a k jeho následnému růstu.

Seznam použité literatury

- [1] *Akcie, burzy & fondy* [online]. 2000 [cit. 2011-06-18]. Miras Lebl personal web. Dostupné z WWW: <<http://www.miras.cz/akcie/>>.
- [2] ALCALÁ-FDEZ, J., et al. *KEEL: a software tool to assess evolutionary algorithms for data mining problems* [online]. [s.l.] : Springer-Verlag, 2008 [cit. 2011-06-01]. Dostupné z WWW: <KEEL: a software tool to assess evolutionary algorithms for data mining problems>.
- [3] BAIRD, L. C. *Residual algorithms: Reinforcement learning with function approximation..* San Francisco, CA : [s.n.], 1995. 37 s.
- [4] BARNARD, E. *Temporal-difference methods and Markov models : IEEE Transactions on Systems, Man, and Cybernetics.* [s.l.] : [s.n.], 1993. 365 s.
- [5] BARTO, A. G. *Reinforcement learning and adaptive critic methods : Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches.* New York. : [s.n.], 1992. 469-491 s.
- [6] BELLMAN, R.; DREYFUS, S. E. *Functional approximations and dynamic programming..* [s.l.] : [s.n.], 1959. 251 s.
- [7] BOYAN, J. A.; MOORE, A. W. *Generalization in reinforcement learning: Safely approximating the value functions..* San Mateo, CA : [s.n.], 1995. 376 s.
- [8] CLOUSE, J.; UTGOFF, P. *A teaching method for reinforcement learning systems..* [s.l.] : [s.n.], 1992. 92-101 s.
- [9] DAYAN, P.; WATKINS, Ch. *Reinforcement Learning* [online]. University College London : Department of Computer Science, 1998 [cit. 2011-06-01]. Dostupné z WWW: <<http://www.gatsby.ucl.ac.uk/~dayan/papers/dw01.pdf>>.
- [10] *Finance.cz* [online]. 2000 [cit.2011-06-18]. Akcie, fondy, dluhopisy, komodity, indexy, data on-line. Dostupné z WWW: <<http://www.finance.cz/kapitalovy-trh/>>.

- [11] GOSAVI, A. *Reinforcement Learning: A Tutorial Survey and Recent Advances* [online]. Journal on Computing : Missouri University of Science and Technology, 2005 [cit. 2011-06-01]. Dostupné z WWW: <<http://web.mst.edu/~gosavia/joc.pdf>>.
- [12] HARMON, M. E.; HARMON, S. S. *Reinforcement Learning: A Tutorial* [online]. Wright State University : [s.n.], 1998 [cit. 2011-06-01]. Dostupné z WWW: <<http://www.nbu.bg/cogs/events/2000/Readings/Petrov/rltutorial.pdf>>.
- [13] JAAKKOLA, T., et al. *Reinforcement learning algorithm for partially observable Markov decision problems..* San Mateo : [s.n.], 1995. 345-352 s.
- [14] JUANG, Ch.; LIN, J.; LIN, Ch. *Genetic Reinforcement Learning through Symbiotic Evolution for Fuzzy Controller Design* [online]. [s.l.] : [s.n.], 2000 [cit. 2011-06-01]. Dostupné z WWW: <http://sci2s.ugr.es/keel/pdf/algorithm/articulo/2000-IEEE_TSMC-Juang-SEFC.pdf>.
- [15] KAELBLING, L. P.; LITTMAN, M. L.; MOORE, A. W. *Reinforcement Learning: A Survey* [online]. USA : Journal of Artificial Intelligence Research 4, 1996 [cit. 2011-06-01]. Dostupné z WWW: <http://www.cs.cmu.edu/afs/cs/project/jair/pub/volume4/kaelbling96a.pdf>
- [16] *Knowledge Extraction based on Evolutionary Learning* [online]. 2004 [cit. 2011-06-15]. Keel. Dostupné z WWW: <<http://www.keel.es/>>.
- [17] LEE, J. W. *Stock price prediction using reinforcement learning* [online]. Pusan , South Korea : Sungshin Women's Univ., Seoul , 2001 [cit. 2011-06-01]. Dostupné z WWW: <http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=931880>.
- [18] *NASDAQ Stock Market* [online]. 205 [cit. 2011-07-14]. NASDAQ Market Indices . Dostupné z WWW: <<http://dynamic.nasdaq.com/dynamic/IndexChart.asp?symbol=IXIC&desc=NASDAQ+COMPOSITE&sec=nasdaq&site=nasdaq&months=36>>.
- [19] OLEJ, V. *Modelovanie ekonomických procesov na báze výpočtovej inteligencie*. 1. vyd. Hradec Králové : Miloš Vognar - M&V, 2003. 160 s. ISBN 80-903024-9-1.

- [20] OLEJ, V.; HÁJEK, P. *Úvod do umělé inteligence : Moderní přístupy*. Vyd.1. Univerzita Pardubice : [s.n.], 2010. 98 s.
- [21] PETŘÍK, P. *Predikce vývoje akciového trhu prostřednictvím technické a psychologické analýzy* [online]. [s.l.], 2010. 59 s. Diplomová práce. Vysoké učení technické v Brně. Dostupné z WWW:<www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=28710>.
- [22] SCHÖNEBURG, E. *Stock price prediction using neural networks: A project report* [online]. 2. [s.l.] : Neurocomputing, 1990 [cit. 2011-06-01]. Dostupné z WWW: <<http://www.sciencedirect.com/science/article/pii/092523129090013H>>.
- [23] SUTTON, R. S. *Generalization in reinforcement learning: Successful examples using sparse coarse coding*. Cambridge : MIT Press, 1996. 1038-1044 s.
- [24] SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning: An Introduction* [online]. Cambridge, Massachusetts : The MIT Press, 1998 [cit. 2011-06-01]. Dostupné z WWW: <<http://neuro.bstu.by/ai/RL-3.pdf>>.
- [25] TAYLOR, G. W. *Reinforcement Learning for Parameter Control of Image-Based Applications* [online]. Ontario, Canada : [s.n.], 2004 [cit. 2011-06-01]. Dostupné z WWW:<http://citeseerx.ist.psu.edu/viewdoc/download?_doi=10.1.1.71.337&rep=rep1&type=pdf>.
- [26] *Wikipedie : Otevřená encyklopedie* [online]. 2006 [cit. 2011-06-15]. Markovův rozhodovací proces. Dostupné z WWW: < http://cs.wikipedia.org/wiki/Markovsk%C3%BD_rozhodovac%C3%AD_proces>.

Seznam obrázků

Obrázek 1 - Interakce agenta s prostředím. Zdroj: [24].....	26
Obrázek 2 - Architektura actor-critic. Zdroj: [24]	35
Obrázek 4 - Kódování TSK typu fuzzy pravidla v chromozomu pomocí symbiotické evoluce. Zdroj: [14].....	37
Obrázek 3 - Kódování v konsekventu fuzzy pravidla s jednoprvkovou množinou v chromozomu pomocí symbiotické evoluce. Zdroj: [14].....	37
Obrázek 5 - Gaussova funkce příslušnosti. Zdroj: [14].....	38
Obrázek 6 - Operace křížení pravidel (jednotlivců) v konsekventní části pravidla. Zdroj: [14].....	40
Obrázek 7 - Blokové schéma navrhované metody konstrukce SEFC. Zdroj: [14].....	41
Obrázek 8 - Návrh modelu pro predikci akciových kurzů. Zdroj: [vlastní]	49
Obrázek 9 - Návrh modelu v prostředí Keel. Zdroj: [vlastní]	53
Obrázek 10 - Softwarový nástroj Keel. Zdroj: [vlastní]	62
Obrázek 11 - Datový management v Keel. Zdroj: [vlastní]	63
Obrázek 12 - Načtení datového souboru. Zdroj: [vlastní]	64
Obrázek 13 - Vizualizace dat. Zdroj: [vlastní]	64
Obrázek 14 - Záložka informací o attributech. Zdroj: [vlastní].....	65
Obrázek 15 - Záložka porovnání atributů. Zdroj: [vlastní].....	66
Obrázek 16 - Importování datové sady a oddílů. Zdroj: [vlastní]	66
Obrázek 17 - Výběr vstupního formátu, nastavení separátoru a nulové hodnoty. Zdroj: [vlastní]	66
Obrázek 18 - Import trénovacích a testovacích souborů. Zdroj: [vlastní]	67
Obrázek 19 - Výběr typu rozdělení dat. Zdroj: [vlastní]	68
Obrázek 20 - Vytvoření jednotlivých oddílů dat. Zdroj: [vlastní]	68
Obrázek 21 - Editace dat. Zdroj: [vlastní]	69
Obrázek 22 - Editace proměnných. Zdroj: [vlastní]	69
Obrázek 23 - Hlavní okno modulu experimentů. Zdroj: [vlastní]	70
Obrázek 24 - Výběr datových souborů pro experiment. Zdroj: [vlastní]	71
Obrázek 25 - Vložení datového uzlu. Zdroj: [vlastní]	72
Obrázek 26 - Přehled metod pro předzpracování. Zdroj: [vlastní]	73
Obrázek 27 - Přehled dostupných metod a algoritmů. Zdroj: [vlastní]	73
Obrázek 28 - Metody pro následné zpracování výsledků. Zdroj: [vlastní].....	74
Obrázek 29 - Statistické testy pro regresi a klasifikaci. Zdroj: [vlastní]	74
Obrázek 30 - Vizualizační moduly pro analýzu a sumarizaci výsledků. Zdroj: [vlastní]	75
Obrázek 31 - Nastavení parametrů algoritmu. Zdroj: [vlastní]	76
Obrázek 32 - Zpráva o procesu generování experimentu. Zdroj: [vlastní].....	76

Seznam grafů

Graf 1 - Vývoj hodnot indexu nasdaq composite. Zdroj: [vlastní].....	43
Graf 2 - Klouzavý průměr $n=5$. Zdroj: [vlastní]	45
Graf 3 - Vážený klouzavý průměr $n=5$. Zdroj: [vlastní].....	46
Graf 4 - Indikátor momentum $n=5$. Zdroj: [vlastní]	47
Graf 5 - Směrodatná odchylka $n=5$. Zdroj: [vlastní]	48

Seznam tabulek

Tabulka 1 - Výchozí nastavení parametrů algoritmu SEFC. Zdroj: [vlastní].....	54
Tabulka 2 - Výsledky genetického algoritmu SEFC pro různé hodnoty N_f . Zdroj: [vlastní]...	55
Tabulka 3 - Výsledky genetického algoritmu SEFC pro různé hodnoty N_r . Zdroj: [vlastní] ..	56
Tabulka 4 - Výsledky genetického algoritmu SEFC pro dané nastavení parametrů. Zdroj: [vlastní]	56
Tabulka 5 - Optimální parametry SEFC. Zdroj: [vlastní].....	57
Tabulka 6 - Výsledky predikce sítě RBF pro různý počet neuronů ve skryté vrstvě. Zdroj: [vlastní]	58
Tabulka 7 - Výsledky predikce MLP pro dané nastavení parametrů. Zdroj: [vlastní]	60
Tabulka 8 - Porovnání výsledků metod. Zdroj: [vlastní].....	61

Seznam příloh

Příloha 1 – Ukázka vstupních dat.....	87
Příloha 2 – Popisné statistiky pro trénovací a testovací data – normalizované vstupy.....	88
Příloha 3 – Ukazatele technické analýzy	89

Přílohy

Příloha 1 – Ukázka vstupních dat

Datum	Index Nasdaq	SMA n=5	WMA n=5	MOM n=5	STD n=5	Výstup o 5 dní
1.5.2007	2434.25	2438.57	2442.310	100.853	11.233	2502.82
1.8.2007	2438.20	2445.81	2435.659	100.240	10.444	2497.78
1.9.2007	2443.83	2452.09	2444.519	102.079	20.646	2479.42
1.10.2007	2459.33	2465.81	2460.849	102.650	27.476	2443.21
1.11.2007	2484.85	2477.72	2484.669	102.208	25.347	2451.31
1.12.2007	2502.82	2484.84	2500.664	100.817	17.109	2431.07
1.16.2007	2497.78	2481.62	2498.999	98.324	23.458	2431.41
1.17.2007	2479.42	2474.91	2473.704	97.942	26.852	2466.28
1.18.2007	2443.21	2460.56	2454.765	97.329	27.374	2434.24
1.19.2007	2451.31	2447.28	2440.890	98.064	19.877	2435.49
1.22.2007	2431.07	2444.66	2434.067	100.944	14.782	2441.09
1.23.2007	2431.41	2442.86	2441.301	99.304	15.527	2448.64
1.24.2007	2466.28	2439.70	2448.997	100.182	14.977	2463.93
1.25.2007	2434.24	2441.70	2445.309	100.398	14.183	2468.38
1.26.2007	2435.49	2445.15	2433.215	99.285	13.110	2475.88
1.29.2007	2441.09	2444.68	2440.388	101.220	12.172	2470.60
1.30.2007	2448.64	2451.51	2450.729	101.350	14.250	2471.49
1.31.2007	2463.93	2459.58	2461.147	101.425	14.349	2490.50
2.1.2007	2468.38	2465.49	2470.927	100.897	10.353	2488.67
2.2.2007	2475.88	2470.06	2472.899	100.307	4.376	2459.82
2.5.2007	2470.60	2475.37	2471.200	100.896	8.886	2450.38
2.6.2007	2471.49	2479.43	2475.854	100.517	9.507	2459.88
2.7.2007	2490.50	2476.22	2487.690	99.564	13.056	2488.38
2.8.2007	2488.67	2472.17	2484.161	99.146	17.579	2497.10
2.9.2007	2459.82	2469.85	2463.840	98.771	18.437	2496.31
2.12.2007	2450.38	2469.43	2450.335	99.988	17.859	2513.04
2.13.2007	2459.88	2471.11	2463.209	101.516	20.353	2518.42
2.14.2007	2488.38	2478.41	2484.176	101.874	21.784	2524.94
2.15.2007	2497.10	2490.94	2495.663	102.161	19.535	2515.10
2.16.2007	2496.31	2502.65	2501.101	101.207	12.563	2504.52
2.20.2007	2513.04	2509.96	2509.495	101.115	12.818	2407.86
2.21.2007	2518.42	2513.56	2520.991	100.753	10.646	2416.15
2.22.2007	2524.94	2515.20	2522.101	99.661	7.48	2404.21
2.23.2007	2515.10	2494.17	2523.754	95.610	48.81	2368.00
2.26.2007	2504.52	2473.71	2480.831	95.691	56.87	2340.68
2.27.2007	2407.86	2449.57	2434.964	95.591	55.29	2385.14
2.28.2007	2416.15	2420.15	2405.767	94.549	50.64	2374.64

Příloha 2 – Popisné statistiky pro trénovací a testovací data – normalizované vstupy

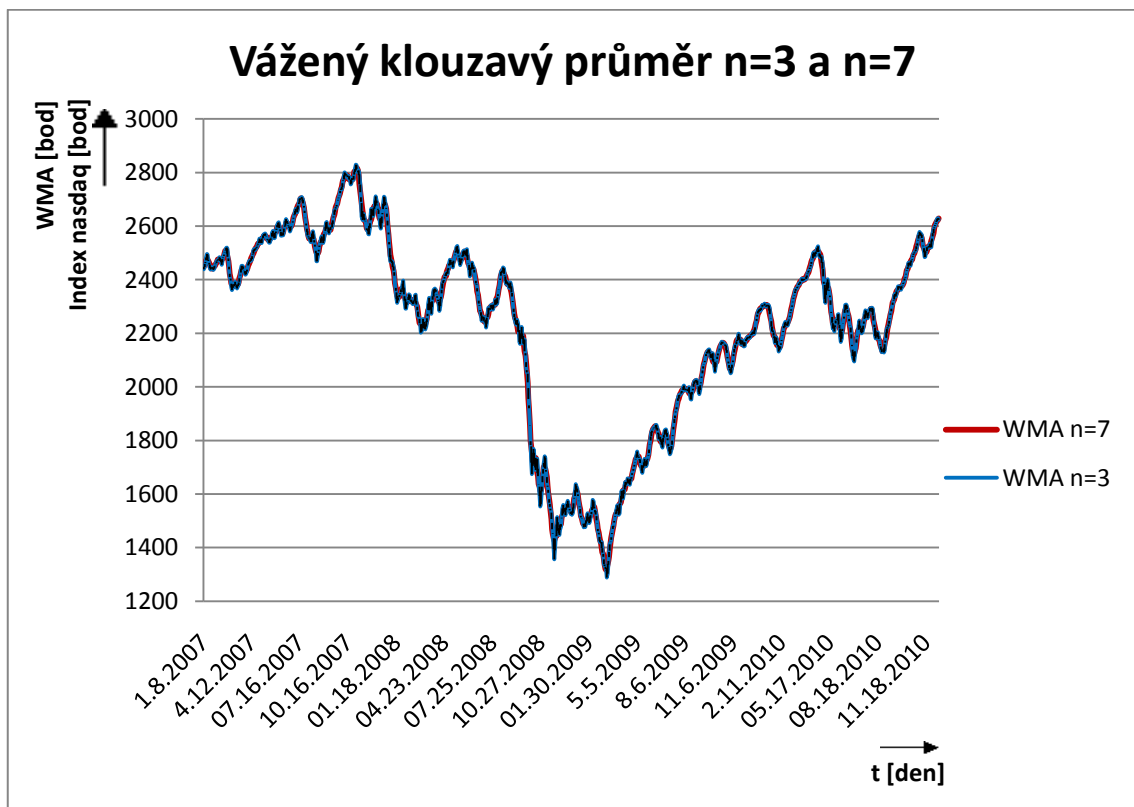
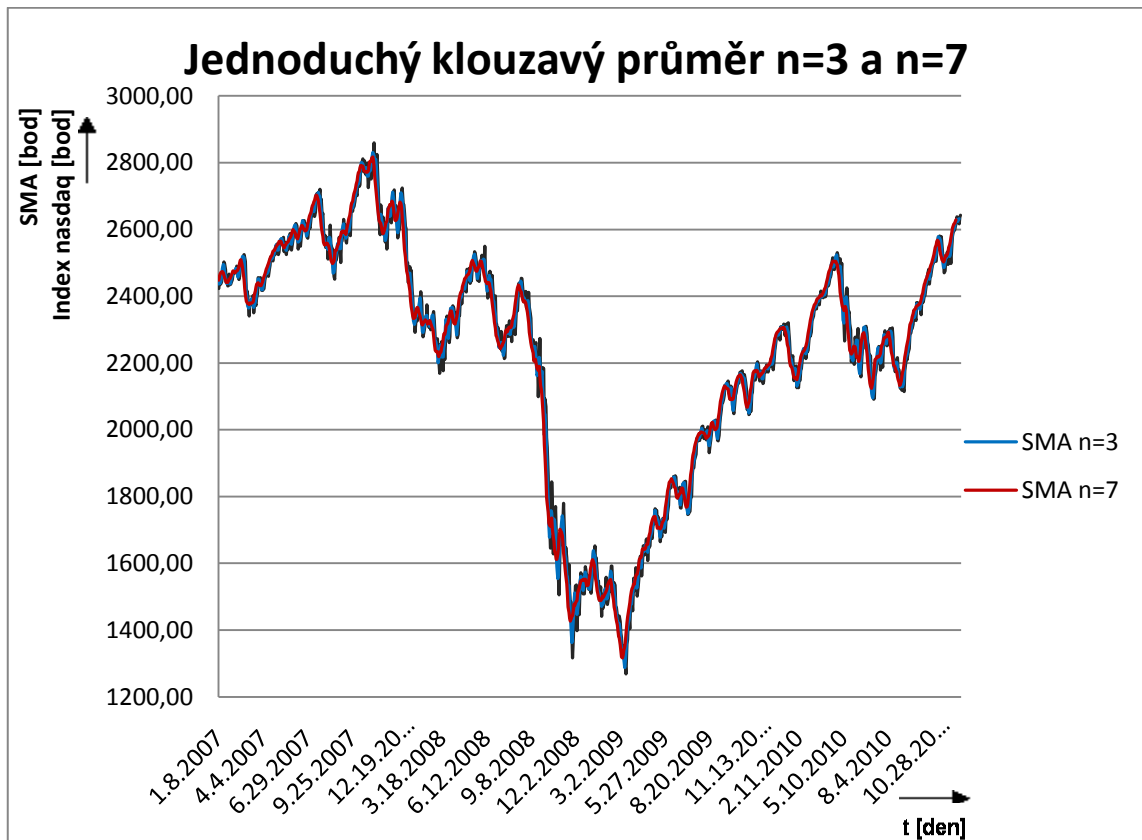
datové oddíly	Průměry pro trénovací data				
	X1	X2	X3	X4	Y
A1	0.791	0.787	0.859	0.240	2231.268
A2	0.789	0.785	0.860	0.243	2224.913
A3	0.789	0.785	0.860	0.242	2226.640
A4	0.792	0.788	0.859	0.240	2235.512
A5	0.791	0.787	0.859	0.242	2228.259
A6	0.790	0.786	0.860	0.239	2228.457
A7	0.790	0.786	0.860	0.243	2228.482
A8	0.789	0.785	0.860	0.241	2224.877
A9	0.789	0.785	0.860	0.241	2225.925
A10	0.789	0.785	0.860	0.242	2225.956

datové oddíly	Směrodatné odchylky pro trénovací data				
	X1	X2	X3	X4	Y
A1	0.015	0.015	0.001	0.019	123978.732
A2	0.016	0.016	0.001	0.019	129230.047
A3	0.016	0.016	0.001	0.019	128865.231
A4	0.016	0.016	0.001	0.019	125917.780
A5	0.016	0.016	0.001	0.020	128085.627
A6	0.016	0.016	0.001	0.017	130561.046
A7	0.016	0.016	0.001	0.019	129681.750
A8	0.016	0.016	0.001	0.019	131397.368
A9	0.016	0.016	0.001	0.019	128444.758
A10	0.016	0.016	0.001	0.019	129443.813

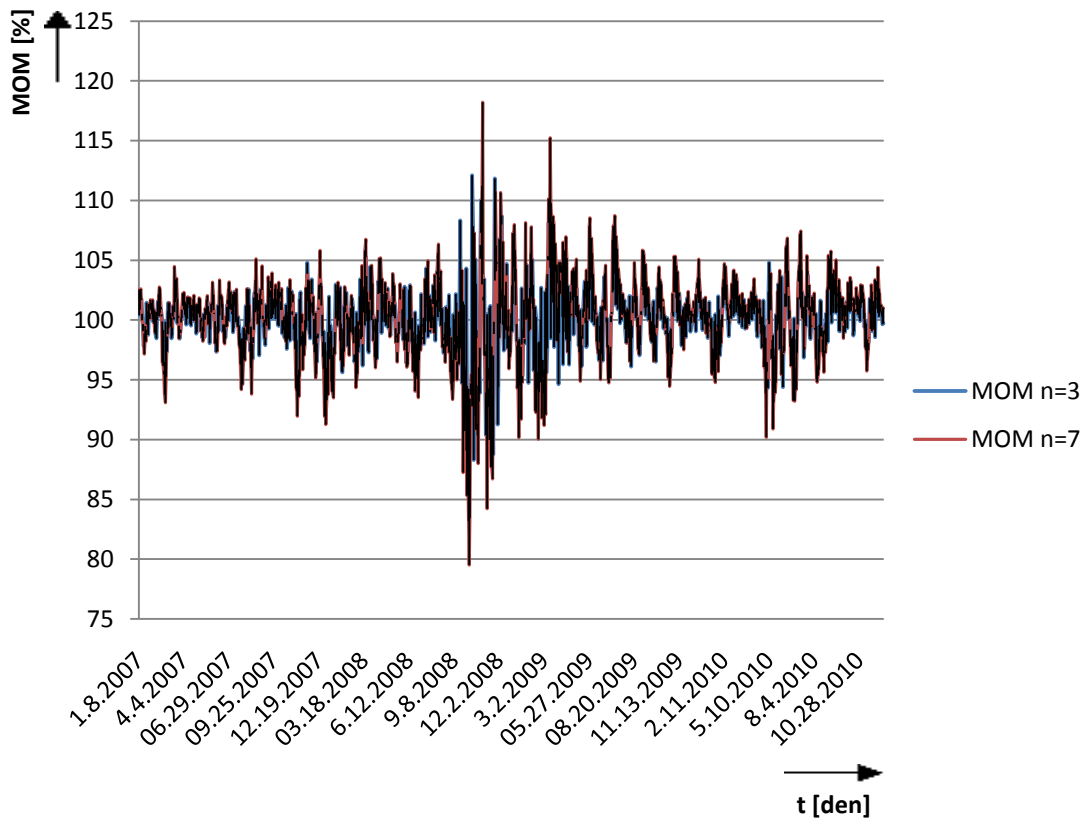
datové oddíly	Průměry pro testovací data				
	X1	X2	X3	X4	Y
A1	0.781	0.776	0.865	0.253	2199.133
A2	0.799	0.796	0.860	0.224	2255.821
A3	0.794	0.791	0.858	0.236	2240.56
A4	0.766	0.763	0.864	0.258	2160.531
A5	0.783	0.777	0.865	0.234	2225.956
A6	0.791	0.787	0.853	0.259	2224.165
A7	0.790	0.786	0.857	0.227	2223.94
A8	0.799	0.794	0.859	0.243	2256.458
A9	0.798	0.794	0.855	0.247	2247.006
A10	0.797	0.793	0.861	0.233	2246.729

datové oddíly	Směrodatné odchylky pro testovací data				
	X1	X2	X3	X4	Y
A1	0.021	0.021	0.001	0.023	170115.737
A2	0.015	0.015	0.001	0.014	122924.151
A3	0.016	0.016	0.001	0.017	126880.715
A4	0.018	0.018	0.001	0.018	148772.031
A5	0.018	0.017	0.001	0.014	134148.085
A6	0.013	0.013	0.001	0.033	111604.796
A7	0.015	0.015	0.001	0.016	119606.172
A8	0.014	0.014	0.001	0.019	103102.342
A9	0.015	0.015	0.001	0.020	130479.935
A10	0.015	0.015	0.001	0.017	121398.203

Příloha 3 - Ukazatele technické analýzy



Momentum n=3 a n=7



Směrodatná odchylka n=3 a n=7

