



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ALGORITMICKÉ OBCHODOVÁNÍ NA BURZE S VYU- ŽITÍM UMĚLÝCH NEURONOVÝCH SÍTÍ

ALGORITHMIC TRADING USING ARTIFICIAL NEURAL NETWORKS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MICHAL CHLUD

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. IGOR SZÓKE, Ph.D.

BRNO 2016

Zadání diplomové práce

Řešitel: **Chlud Michal, Bc.**

Obor: Počítačová grafika a multimédia

Téma: **Algoritmické obchodování na burze s využitím umělých neuronových sítí**

Algorithmic Trading Using Artificial Neural Networks

Kategorie: Umělá inteligence

Pokyny:

1. Seznamte se s teorií umělých neuronových sítí a intraday obchodování na burze.
2. Prostudujte dostupné techniky technické analýzy.
3. Seznamte se s vybraným nástrojem pro trénování neuronových sítí.
4. Definujte vhodná data pro trénování a testování metod pro algoritmické obchodování.
5. Zvolte vhodné indikátory technické analýzy či jiné parametry pro vstup neuronové sítě. Natrénujte a otestujte základní obchodní algoritmus.
6. Upravte výběr popř. vhodně zkombinujte indikátory a zlepšete Váš základní obchodní algoritmus.
7. Zhodnoťte výsledky, vyberte nejvhodnější metodu (parametry modelu a indikátory) a navrhněte směry dalšího vývoje.
8. Vytvořte A2 plakátek a cca 30 vteřinové video prezentující výsledky vašeho projektu.

Literatura:

- Dle pokynů vedoucího

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 5 ze zadání.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci dřívějších projektů (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Szóke Igor, Ing., Ph.D.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2015

Datum odevzdání: 25. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Tato diplomová práce se zabývá algoritmickým obchodováním na burze s využitím umělých neuronových sítí. V první části jsou popsány základní termíny týkající se obchodování na burze a algoritmického obchodování, také je zde k dispozici teoretický úvod do neuronových sítí. V druhé části jsou specifikována data, na kterých bude probíhat simulace obchodování. Na těchto datech se také učí neuronová síť. Ta je využita pro predikci budoucí hodnoty trhu v automatické obchodní strategii. Ke konci je navzájem porovnáno několik strategií s různými variantami neuronových sítí.

Abstract

This diploma thesis deals with algorithmic trading using neural networks. In the first part, some basic information about stock trading, algorithmic trading and neural networks are given. In the second part, data sets of historical market data are used in trading simulation and also as training input of neural networks. Neural networks are used by automated strategy for predicting future stock price. Couple of automated strategies with different variants of neural networks are evaluated in the last part of this work.

Klíčová slova

Algoritmické obchodování, obchodování na burze, neuronová síť, predikce časové řady, technická analýza, strojové učení

Keywords

Algorithmic trading, stock trading, neural network, time series prediction, technical analysis, machine learning

Citace

CHLUD, Michal. *Algoritmické obchodování na burze s využitím umělých neuronových sítí*. Brno, 2016. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Szóke Igor.

Algoritmické obchodování na burze s využitím umělých neuronových sítí

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Igora Szókeho Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Michal Chlud
25. května 2016

Poděkování

Tímto bych chtěl poděkovat svému vedoucímu za pomoc a vedení při tvorbě této práce.

© Michal Chlud, 2016.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	3
2 Obchodování na burze	4
2.1 Co obchodovat na elektronické burze	4
2.1.1 Akcie	4
2.1.2 Komodity	4
2.1.3 E-mini futures	5
2.1.4 Forex	5
2.1.5 Opce	5
2.2 Obchodování v praxi	6
2.2.1 Volatilita	6
2.2.2 Likvidita	6
2.2.3 Trend trhu	6
2.2.4 Grafy	6
2.2.5 Spekulace na růst nebo pokles	8
2.2.6 Brokeři a data	9
2.3 Analýza trhu	9
2.4 Obchodní styl - timeframe	10
3 Algoritmické obchodování	12
3.1 Obchodní signály	13
3.1.1 Indikátory trendu	13
3.1.2 Indikátory volatility	15
3.1.3 Oscilátory	15
3.2 Co by měla strategie umět	16
3.3 Zhodnocení úspěšnosti algoritmu	16
4 Neuronové sítě	19
4.1 Backpropagation	21
4.2 Učení sítě	22
4.3 Aplikace neuronových sítí	23
5 Návrh vlastní obchodní strategie	25
5.1 Použitá obchodní data	25
5.2 Testování obchodní strategie	28
5.3 Obchodování pomocí neuronové sítě	29
5.3.1 Předzpracování vstupních dat	30
5.4 Implementace	32

6	Vyhodnocení regresní obchodní strategie	33
6.1	Referenční obchodní strategie	33
6.2	Regresní neuronová síť	36
6.2.1	Výsledky pro učení bez indikátorů	38
6.2.2	Optimalizace použitých indikátorů	41
6.2.3	Výsledky pro učení s indikátory	42
7	Kombinace klasifikační a regresní sítě	45
7.0.1	Vstupy a topologie neuronové sítě	47
7.0.2	Úprava obchodní strategie	48
7.0.3	Učení klasifikační sítě	49
7.0.4	Učení klasifikační sítě na TF	53
7.1	Optimalizace parametrů obchodní strategie	58
7.1.1	Optimalizace pomocí PSO na EMD	58
7.1.2	Výsledky pro optimalizovaný obchodní model na EMD	61
8	Živé obchodování	62
8.1	Výsledky	63
9	Závěr	66
	Literatura	68

Kapitola 1

Úvod

Algoritmické obchodování vzniklo jako přirozená reakce na příchod burzovního obchodování do elektronického a internetového světa. Výpočetní systémy nemusí být používány pouze jako nástroj k analýze. Lze zkonstruovat automatizované obchodní strategie, které provádí obchodování autonomně. Výhodou takového automatizovaného systému oproti člověku je především jeho odolnost vůči psychologickým vlivům. Je schopen rozhodovat se a podávat příkazy daleko rychleji než lidský obchodník. Obchod nemusí trvat minuty, ale pouze desítky mikrosekund. Tyto možnosti vedly k novým trendům jako je třeba vysokofrekvenční obchodování.

Algoritmický systém obchoduje pouze tak, jak je naprogramován. To především znamená, že je konzistentní. Před nasazením systému probíhá testování implementované strategie. Systém je důkladně testován na historických a aktuálních datech, tak aby byly odhaleny jeho slabiny. Algoritmický obchodní systém většinou obsahuje řadu parametrů pomocí nichž může být jeho úspěšnost optimalizována.

Metody strojového učení jsou vhodnými kandidáty pro aplikaci v algoritmickém obchodování. Konkrétně prediktivní a klasifikační schopnosti neuronových sítí do této oblasti vhodně zapadají. Pokud platí předpoklad, že chování trhu není čistě náhodné a že v něm lze rozpoznat určité opakující se jevy, tak je možné, že se při vhodně zvolených trénovacích datech bude schopna neuronová síť něco o chování trhu naučit. Algoritmická obchodní strategie může pak využít predikce k rozhodování o tom, zda vstoupit nebo vystoupit z určitého obchodu.

Tato práce se v teoretické části zabývá některými základními termíny a možnostmi, které se týkají obchodování na burze. Jde například o různé obchodovatelné burzovní produkty a jejich vlastnosti, metody technické analýzy, typy technických indikátorů, způsoby zhodnocení úspěšnosti algoritmu a v poslední řadě neuronovými sítěmi.

V druhé části jsou zprvu specifikována trénovací a testovací obchodní data a dále se práce zabývá návrhem a vyhodnocením jednoduché obchodní strategie, která využívá regresní neuronovou síť k predikci budoucí ceny trhu. Dále pak návrhem pokročilé obchodní strategie, která kombinuje předchozí regresní neuronovou síť a klasifikační neuronovou síť. V poslední části je popsán experiment s reálným obchodováním na demo účtu.

Kapitola 2

Obchodování na burze

Burza je vysoce organizovaná forma trhu. Do dnešní podoby dospěla dlouhodobým vývojem. Trhy byly původně neorganizované a nepravidelné. Docházelo na nich ke směně zboží. Později se začaly organizovat. Už ve středověku muselo město vlastnit právo na trh, protože byl zdrojem vysokého příjmu. Koncept burzy se objevuje v době, kdy na trh nastupuje nějaká forma zastupitelného zboží jako směňky nebo mince. [10]

V dnešní době už neplatí představa o burze jako o hektickém místě, kde spousta brokerů nabízí nebo poptává akcie. S přechodem burzy na elektronický systém vznikla pro spoustu lidí možnost obchodovat osobně ze svého domova. Zavedením elektronického systému bylo také umožněno využití počítačů a obchody lze v současnosti rušit a uzavírat mnohem rychleji než dříve.

2.1 Co obchodovat na elektronické burze

Na burze je možné obchodovat několik produktů. Navzájem se liší rizikovostí, likviditou a dalšími svými vlastnostmi. [14]

2.1.1 Akcie

Jedním z neznámějších a typicky spojovaným produktem s burzou jsou akcie. Obecná definice zní tak, že akcie jsou obchodovatelné cenné papíry. Tržní cena akcie (kurz akcie) je cena, kterou se akcie obchoduje na trhu. Akcie jsou nakupovány za účelem zisku. Pokud cena akcie vzroste a obchodník je vhodně prodá, tak tím generuje zisk.

Okamžitou hodnotu akcie reprezentuje aktuální kurz na burze cenných papírů. Na hodnotu akcie mají vliv hospodářské výsledky společnosti, stavu ekonomiky, konkurence a tak dále. Kurz není regulován uměle, ale je stanoven nabídkou a poptávkou po daných akciích.

2.1.2 Komodity

Komodity představují základní suroviny. Na komoditní burze nejsou přímo obchodovány tyto suroviny, ale jejich kontrakty. Obchodník se může rozhodnout pro obchodování zlata, ropy, bavlny, kaka, pšenice, zemního plynu a podobně. Jde o tzv. futures kontrakty. Vývoj ceny kontraktů lze spekulovat na základě mezinárodní politiky a stavu globálního hospodářství. Fundamentální zprávy mají vliv na vývoj cen (odhady úrody, předpověď exportů, velikost produkce).

Komodity jsou na trhu populární. Za lákavostí komodit stojí požadavky na blokovanou zálohu, které jsou velmi nízké. I s poměrně malou sumou lze kontrolovat několikanásobně vyšší pozici, říká se tomu tzv. páka (leverage). Poměr u komoditních obchodů se pohybuje většinou v rozmezí 1:5 až 1:20. Investor může například vstoupit do obchodu s částkou 100\$, ale páka mu umožní kontrolovat pozici s částkou 1000\$.

2.1.3 E-mini futures

Jde o standardizovaný burzovní finanční derivát. Z důvodu zpřístupnění futures kontraktů větší části veřejnosti zavedly burzy zmenšené kontrakty. ‚E‘ v názvu znamená, že tyto kontrakty jsou obchodovány pouze elektronicky na elektronických burzách. Výhodou je, že obchody mohou probíhat 24h denně, na rozdíl od klasických obchodů, jež jsou vázány otevírací dobou burzy.

Nejpopulárnější E-mini futures existují na akciové indexy (E-mini S&P 500, E-mini Russel 2000). Existují také futures na bondy, měny (E-mini Euro/USD) a komodity (E-mini na ropu nebo zlato). Výhodou toho typu obchodů je především velká páka (leverage) a vysoká likvidita trhu.

2.1.4 Forex

Forex je obchodování s měnami [5]. Forex je trh vhodný i pro investory s malým počátečním kapitálem. Jednotlivé měny se obchodují vždy v tzv. měnových párech (např. EUR/USD). U každého páru je jako první zobrazena základní měna. V případě, že základní měna z páru sílí, tak hodnota měnového páru stoupá. Pokud měnový pár oslabuje, tak je to naopak.

Původně byl forexový trh přístupný pouze malému počtu investorů. To se změnilo s příchodem internetu, v té době vznikly brokeri, kteří nabízejí přístup veřejnosti k těmto trhům. Obchodování probíhá na rozdíl od akcií 24 hodin denně.

V tomto typu obchodu se také uplatňuje pákový efekt. Obchodník může s vkladem 50\$ manipulovat s měnou v hodnotě až 20 000\$.

Brokeri dostávají zaplacen v podobě tzv. spreadu. Při volbě správného brokera se náklady na obchody pohybují mezi nejnižšími z celého trhu.

2.1.5 Opce

Opce (options) se začaly intenzivně využívat od 70. let. Opce jako podmíněné deriváty představují termínové kontrakty, v nichž držitel opce (kupující, holder) má právo uskutečnit ve sjednaném termínu příslušný obchod. Vstup do obchodu se uskuteční zakoupením opce za její cenu. Opce se obchodují v kontraktech a podobně jako komodity mají také dobu vypršení, takže není možné je držet navždy (jako akcie).

Mechanismus nákupu a prodeje je stejný jako u jiných produktů řízen pomocí nabídky a poptávky. Opce vycházejí vždy ze svého podkladového aktiva. Lze obchodovat opce, jejichž základem je měna, akcie, akciový index nebo například komodity (opce na Microsoft, Coca-Colu, cukr, zlato, dolary a eura).

Opět se zde uplatňuje pákový efekt, kdy je možné kontrolovat podkladové aktivum za zlomek jeho ceny.

2.2 Obchodování v praxi

Existuje spousta obchodovatelných produktů, kterými se zabývá velký počet burz po celém světě. Liší se od sebe různými vlastnostmi, jako je likvidita, rizikovitost investice nebo doba obchodování, velikost páky (leverage) a tak dále.

Pro zopakování tzv. **páka** znamená, že obchodník může složením relativně malé zálohy ovládat mnohem větší částku na trhu. Z toho vyplývá, že může získat daleko větší výdělek. Nevýhodou je naopak to, že stejně rychle může o svou investici přijít. Velikost páky se pohybuje od 1:2 až do 1:100. Tato hodnota je předem stanovena brokerem.

Akcie je charakterizována svou aktuální cenou na trhu. Termín „tick“ souvisí se změnou ceny. Je to nejmenší bodová jednotka obchodu, o kterou se může cena změnit. Od roku 2001 je minimální velikost ticku 1 cent (platí pro akcie, které se obchodují za cenu vyšší než 1\$). Futures kontrakty mají tuto velikost proměnlivou a uvádí se u daného obchodu. Obdoba ticku u forexových trhů je pip.

2.2.1 Volatilita

Volatilita je statistická míra, která označuje míru kolísání aktiva [10]. Většinou se určí pomocí směrodatné odchylky. Důležitým parametrem volatility je, za jak dlouhé období je vypočítána. To by mělo odpovídat časovému měřítku (timeframe), v jakém chce obchodník investovat. Hodnota volatility aktiva souvisí s rizikovostí obchodu. Trh s větší volatilitou umožňuje větší výdělky, ale také rychlejší ztrátu.

2.2.2 Likvidita

Vysoká likvidita trhu znamená, že existuje velká šance, aby byl požadovaný obchod (prodej, nákup) uskutečněn za aktuální požadovanou cenu. Opačně nízká likvidita znamená, že šance je malá.

2.2.3 Trend trhu

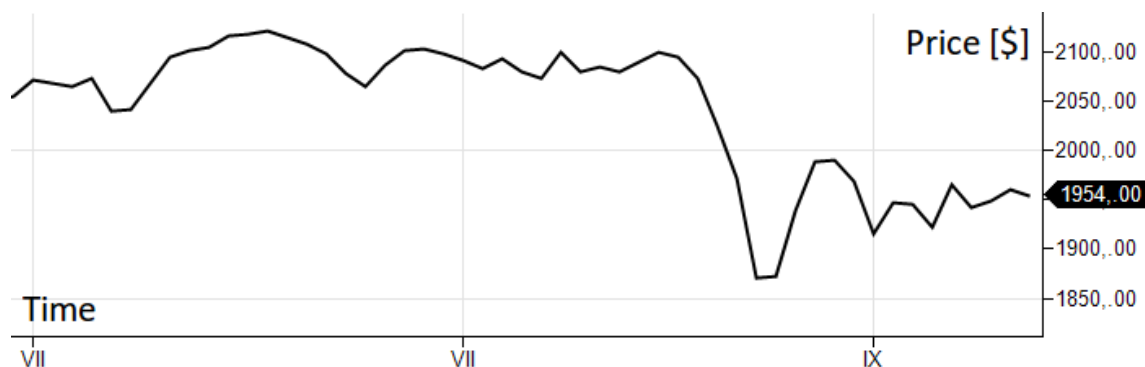
Trend je dlouhodobá změna trhu jedním směrem. Trend může mít charakter buď rostoucí nebo klesající. Rostoucímu trendu se říká tzv. býčí trh (bull market). Klesajícímu trendu se říká medvědí trh (bear market). Trhům, které nemají jasný trend se říká, že jdou tzv. do strany.

2.2.4 Grafy

Zakreslením ceny vybraného aktiva do grafu získává obchodník přehledný a snadno čitelný nástroj. Grafy reprezentují pohyb ceny v čase. Grafická reprezentace zjednodušuje orientaci a pomáhá určit trend, kterým se cena ubírá [8]. Je to základní prvek technické analýzy, jejíž podstatou je snaha o odhad dalšího vývoje na základě cen z minulosti.

S grafy souvisí volba časového rozsahu (timeframe) neboli rozlišení grafu. Cena se může pohybovat s nejmenším možným rozdílem – tickem. Pokud má být vývoj ceny zobrazen v čase, tak musí být zvoleno vhodné časové měřítko.

Například jeden bod v **čarovém grafu** reprezentuje zavírací cenu (close price) v daném časovém okamžiku. Je to nejjednodušší typ grafu a má také nejmenší vypovídací hodnotu o tom, co se na trhu děje.



Obrázek 2.1: Čarový graf (NinjaTrader)

Lze z něj určit trend trhu. Zda roste, klesá nebo se pohybuje do strany.

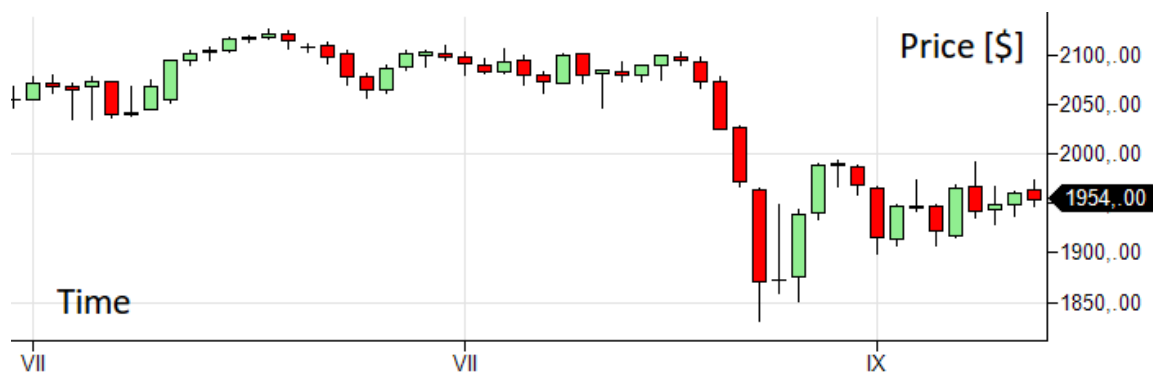
Při reprezentaci vývoje cen v grafu se používají termíny HIGH – nejvyšší cena za dané časové období. LOW – nejnižší cena za dané časové období. OPEN – cena při otevření trhu nebo v počátku časového období. CLOSE – závírací cena nebo cena na konci časového období.

Sloupcový graf má lepší vypovídací hodnotu, protože zachycuje výše zmíněné hodnoty. Jednotlivé čárky představují obchodní rozpětí ve vybraném časovém úseku (minuta, hodina, den, měsíc apod.) Vrchol čárky je nejvyšší cena (HIGH) daného období. Spodek čárky je nejnižší cena za dané období (LOW). Výběžek na levé straně je otevírací cena (OPEN) a výběžek na pravé straně je závírací cena (CLOSE).



Obrázek 2.2: Sloupcový graf (NinjaTrader)

Svíčkový graf [8]. Tento typ grafu vznikl na základě metodologických postupů Muneshia Homma, který v 18. století v Japonsku zkoumal obchod s rýží. Začal se používat technickými analytiky s rozvojem japonského akciového trhu v 70. letech 19. století. Svíčkový graf reprezentuje stejné hodnoty jako čárkový graf, ale jeho výhodou je, že je na rozdíl od něj přehlednější. Plná svíčka vyjadřuje pokles v ceně za dané období a prázdná svíčka nárůst. Existují metody, jak v grafu rozpoznávat tzv. reverzní signály, které značí obrat v trendu trhu apod.



Obrázek 2.3: Svíčkový graf (NinjaTrader)

2.2.5 Spekulace na růst nebo pokles

Spekulace na vzestup ceny je nejpoužívanější strategií. Obchodník nakoupí aktiva, protože spekuluje s růstem ceny v budoucnu, aby pak mohl prodat za cenu vyšší než nákupní. Umění také je odhadnout, kdy nakoupená aktiva prodat.



Obrázek 2.4: Vstup do dlouhé pozice (NinjaTrader)

Obchodník vstupuje do **dlouhé pozice** (anglicky long), protože počítá s nárůstem ceny. Nakoupil za 1854 USD 10 akcií. Po nějaké době prodá akcie za cenu 2000 USD. To znamená zisk 1460 USD (bez poplatků).

Vstupem do **krátké pozice** (short) se označuje naopak spekulace na pokles ceny. Obchodník si vypůjčí akcie (nebo jiný burzovní produkt) od brokera a prodá je na trhu. Pokud následně dojde k poklesu ceny akcie na trhu, tak je ve vhodnou chvíli koupí zpět za sníženou cenu, aby je mohl vrátit brokerovi. Tím generuje zisk a brokerovi odevzdá určitý poplatek.



Obrázek 2.5: Vstup do krátké pozice (NinjaTrader)

2.2.6 Brokeři a data

S akciemi se obchoduje na burze cenných papírů. Přímý přístup k obchodu mají pouze členové burzy, tzv. brokerské společnosti (brokeři). Členské místo v burze bývá také předmětem obchodu. Cena místa u nejobchodovanějších burz se může vyšplhat až na několik milionu dolarů.

Pokud chce uživatel získat přístup k trhu, musí k tomu využít prostředníka – tzv. brokera. Ten na trhu vykonává jeho příkazy jako koupit/prodat a za tuto službu si účtuje určité poplatky. Brokerů existuje velká spousta a liší se podle zaměření burzovního produktu, dalším parametrem je také minimální velikost vkladu. Velká část brokerů umožňuje vyzkoušení služeb na tzv. demo účtu. Uživatel dostane přístup na trh s fiktivním kapitálem.

Kvalitní broker by měl nabízet propracovanou obchodní platformu, korektní data, analytické nástavby, rychlý přístup a nízké poplatky. Základem je kvalitní technická podpora dostupná 24 denně.

Z pohledu algoritmického obchodování je velmi důležité kvalitní API. Knihovna bývá většinou naprogramována v C++ nebo C#. Přes API je možné také získat přístup k datům. Broker většinou nabízí přístup v menším objemu za 10 – 20 USD měsíčně. Nejcennější jsou ticková data. Ty z pohledu testování a experimentování poskytují přesnou reprezentaci toho, co se na trhu dělo. Balíky těchto dat se dají koupit za několik stovek až tisíců dolarů. Dostupnější jsou data s minutovým, hodinovým nebo denním rozlišením. Vhodnost těchto dat závisí na časovém rámci obchodníka.

2.3 Analýza trhu

Obchodník může vstupem do krátké nebo dlouhé pozice dosáhnout zisku, pokud správně odhadne trend trhu a dobu kdy do pozice vstoupit a kdy z ní vystoupit. Jde o spekulaci nad vývojem ceny. Existují dva základní přístupy k analýze trhu. Prvním typem je analýza fundamentální a druhým technická.

Fundamentální analýza

Cílem fundamentální analýzy je správně odhadnout vývoj trhu na základě dlouhodobého pozorování. Fundamentální obchodník se většinou soustředí na několik málo trhů, o kterých

má dobrý přehled. Zajímá se o zprávy, které se tohoto trhu týkají. V případě komoditních trhů by šlo třeba o předpovědi počasí, ekonomické a hospodářské zprávy z daného sektoru a informace, které se týkají velkých firem v dané oblasti. Na základě takových informací se bude snažit předpovědět reakci ostatních obchodníků a tím i vývoj ceny. Obchodníci mohou své pozice držet i několik let.

Technická analýza

Protipól tvoří technická analýza. Fundamentální analýza není schopná předpovídat změny cen v rozsahu např. několika hodin. K tomu se používá právě technická analýza. Ta je vhodná k určení přesnějšího načasování obchodu. Technická analýza, na rozdíl od fundamentální analýzy, využívá historii cen a objemů obchodu. Předpokládá, že trh se řídí nějakými pravidly a opakují se v něm určité vzorce [2]. Ceny jsou určovány psychologíí davu. Na základě tohoto předpokladu existuje spousta systémů, které se snaží najít v historii nějaké vzorce, jež předpoví budoucí vývoj.

Základem technické analýzy jsou často grafy, ve kterých lze nalézt určité vzory. Mezi další nástroje patří technické indikátory. Pomocí kombinace těchto indikátorů a historie cen je možné vytvořit systémy (obchodní strategie), které napovídají kdy vstoupit do pozice nebo z ní vystoupit. Takové systémy patří k „know-how“ každého obchodníka, ale dají se i koupit. Cena takového systému se může pohybovat ve stovkách dolarů. Klíčovým parametrem pro analýzu je volba časového rámce. Analýza by měla být založena na několika časových rámcích, protože analýza založená pouze na denním rámci může ztratit přehled o větším trendu (např. týdenním).

Zpravidla se říká, že analýza by měla být založena na třech rámcích. Tento počet by měl být dostačující. [14] Pokud obchodník drží pozici průměrně jednu hodinu, tak by první rámec měl odpovídat hodinovým grafům. Druhý rámec by měl odpovídat 1/4 této hodnoty, tedy 15 minutovému rámci a třetí rámec by měl být analogicky 4 násobek první hodnoty – to znamená 4 hodinovým nebo denním grafům. Obchod by měl být založen na trendu nejvyššího rámce, protože existuje velká pravděpodobnost, že ten se tak rychle nezmění. V takovémto krátkém rozsahu obchodování se většinou neuplatní fundamentální analýza.

2.4 Obchodní styl - timeframe

Obchodních stylů existuje několik, největší rozdíl je ve zvoleném časovém rámci (time frame). Volba obchodního stylu závisí na preferencích obchodníka, od toho se odvíjí typ zvolené analýzy [3].

Investování

Obchodník se bude snažit odhadnout dlouhodobé trendy. K tomu může využít technické i fundamentální analýzy. Pozice lze držet i několik týdnů až roků. Investor musí disponovat dostatečnými finančními prostředky, aby nemusel pozici opustit při nevhodném krátkodobém vývoji. Nutností pro tuto strategii je dostatečný počet fundamentálních i technických dat, z nichž bude vyvozován budoucí vývoj.

Swing

Swing obchodování je styl snažící se rozpoznat signály, které ukazují na změnu ve vývoji ceny. Pozice jsou drženy jeden až čtyři dny. Takový obchodníci využívají zpravidla technické

analýzy. Klíčové je určit vhodné hodnoty stop-lossu, aby nedocházelo k opuštění pozice před tím, než se trend otočí požadovaným směrem. Nejsou tak náročné jako strategie pracující s menším časovým rámcem.

Denní obchodování (intraday)

Strategie charakteristická tím, že pozice nejsou drženy přes noc. Koncept je podobný jako u swing obchodování. Základem je technická analýza a rozpoznání signálů, že dojde k nějakému zajímavému vývoji nebo obratu ceny. Z toho vyplývá i velikost rámce. Denní obchodníci se budou snažit využít krátkodobých fluktuací cen.

Scalping

Scalp obchodování (scalpování). Efektivní strategie na trzích s velkou volatilitou (například E-Mini Russell 2000). Obchodník je v pozici pouze několik minut nebo sekund. V tak malých časových intervalech se můžou projevit časové prodlevy související s vykonáním daného příkazu (buy, sell).

Tento typ obchodování je velmi náročný na psychiku a vyžaduje velké soustředění. Investor se snaží vydělat na krátkých fluktuacích ceny.

High frequency trading

Čistě automatická forma obchodování. Pozice jsou drženy v řádu milisekund. Člověk už není schopný v tak krátkých intervalech reagovat a naprogramovanou strategii vykonává stroj. V tomto typu obchodu je klíčová rychlost vykonání obchodního příkazu, proto si firmy používající tuto strategii pronajímají servery v co největší blízkosti burzy, aby si zajistili co nejkratší odezvu oproti zbytku trhu. Využívá se toho, že taková strategie reaguje rychleji než trh, finanční objem obchodu nemusí být velký. Přestože nejsou objemy velké, tak jsou tyto transakce vykonávány s velkou frekvencí a mohou generovat velké zisky. Vyžadují však kvalitní technické zázemí a propracované algoritmy. Z toho vyplývá, že nejsou úplně dostupné pro běžnou veřejnost.

Kapitola 3

Algoritmické obchodování

Algoritmické obchodování je automatické. Počítačový systém má na starosti obchodní rozhodnutí a jejich vykonání. Asi do půlky minulého století se nedalo mluvit o nějakých silnějších tendencích vědeckého výzkumu v oblasti financí. Úspěšnému obchodníkovi byla spíše přisuzována dobrá intuice a rozsáhlé zkušenosti. Dnes jsou disciplíny jako ekonomika, matematika, statistika a počítačová věda silně propojeny. Matematické modely mají v současné době na ekonomiku a obchod velký vliv. I burzovní trhy se staly objektem zájmu vědeckého výzkumu, za účelem porozumění mechanismům a principům, které ovládají trh.

Algoritmické obchodování bylo dříve nemožné kvůli technickým omezením. Lze sledovat trend postupného snižování doby, po kterou jsou obchodníci v určité pozici (viz high frequency trading). Algoritmy měli největší úspěch, když se objevili poprvé na trhu koncem minulého století, protože oproti lidem měli značnou výhodu v rychlosti rozhodování a vykonání transakce.

Mezi výhody tohoto typu obchodování určitě patří nezávislost algoritmu na psychice, která může mít na obchodníka, provádějícího rozhodování ručně, silně negativní vliv. Cílem je právě odstranění lidského faktoru z procesu, protože člověk může být se svými výkonech velmi nekonzistentní. Jak už bylo řečeno, velkou výhodou tohoto typu obchodování je, že je možné držet pozice pouze po dobu milisekund.



Obrázek 3.1: Flash Crash 2010 (money.cnn.com)

Algoritmy (strategie) jsou testovány na historických datech a pokud vykazují dostatečné

výsledky, tak bývají před nasazením testovány přímo na živých datech. Klíčové v této oblasti je právě vyhodnocení úspěšnosti a dalších metrik algoritmické strategie. Algoritmy lze také optimalizovat pomocí různých optimalizačních metod jako jsou třeba genetické algoritmy. Mezi efekty způsobené algoritmickým obchodováním na trh patří větší likvidita, rychlejší změna ceny v reakci na nějakou fundamentální událost a větší volatilita.

Mezi negativní dopady algoritmického obchodování se často řadí takzvaný fenomén „flash crash“¹. Nejznámější se stal v roce 2010. Trh se propadl zhruba o trilion dolarů a propad trval přibližně 36 minut. Mezi možné vysvětlení patří, že algoritmické systémy podobně reagovaly na nějakou událost a navzájem se dostaly do „sell“ spirály, která stlačila cenu velmi nízko. Takových bleskových propadů se od té doby stalo více.

3.1 Obchodní signály

Algoritmické obchodování bude zpravidla založeno na technické analýze. Existují i systémy, které jsou postaveny na fundamentální analýze. Jenže nejsou tak rozšířené z důvodu toho, že pro počítače je daleko náročnější zpracovávat a analyzovat sentiment různých informací a zpráv dostupných na internetu. Následuje výpis použitelných technických indikátorů z oboru technické analýzy. Na indikátoru nebo jejich kombinaci může být postavena strategie implementovaná algoritmickým obchodním systémem.

3.1.1 Indikátory trendu

Klouzavý průměr

Klouzavý průměr (Simple Moving Average) patří mezi nejzákladnější indikátory. Je to jednoduchá metoda, která napovídá, kdy do pozice vstoupit nebo z ní vystoupit. Za tímto účelem je vypočítávána průměrná hodnota ceny za určité stanovené období N. Používají se několika způsoby. Prvním způsobem je porovnání klouzavého průměru s aktuální cenou, druhým způsobem je křížení klouzavých průměrů a třetím je seřazení klouzavých průměrů [4].



Obrázek 3.2: Klouzavý průměr s délkou 60 dnů. Vertikální čáry oddělují dny (finance.yahoo.com)

Porovnávání s aktuální cenou funguje dobře v trendových trzích, tedy trzích, které mají tendenci držet se dlouhodobě stejného trendu. Idea je taková, že trh má určitou průměrnou

¹Golub, High Frequency Trading and Mini Flash Crashes, 2012, arXiv:1211.6667

hodnotu a pokud cena vzroste nad tuto hodnotu, tak pod ní za nějakou dobu zase klesne, aby si zachoval průměr. Pokud cena stoupne nad klouzavý průměr, tak je tím generován signál pro nákup. Pokles ceny pod průměrnou hodnotu znamená prodej. Problematické je určit, jak dlouhý interval pro výpočet klouzavého průměru použít. Příklad je na obrázku, kde po protnutí s klouzavým průměrem dojde k růstu trhu a po opětovném protnutí k jeho poklesu.

$$SMA = \frac{P_t + P_{t-1} + P_{t-2} + \dots + P_{t-(N-1)}}{N} \quad (3.1)$$

Jinou variantou jednoduchého klouzavého průměru je exponenciální klouzavý průměr. Jde o váhovaný klouzavý průměr, kde váhy vzorků klesají exponenciálně. Největší váha je na nejaktuálnějším vzorku.

MACD

MACD je jedním z nejčastěji používaných indikátorů. V klasickém nastavení se jedná o rozdíl mezi 26 denním a 12 denním exponenciálním klouzavým průměrem (EMA). Nad rozdílem těchto EMA je většinou zanesen ještě 9denní EMA, této křivce se říká signální. Indikátor se dá aplikovat jako signál k nákupu pokud signální křivka protne zespodu MACD.

$$MACD_t(12, 26) = EMA_t(12) - E_t(26) \quad (3.2)$$



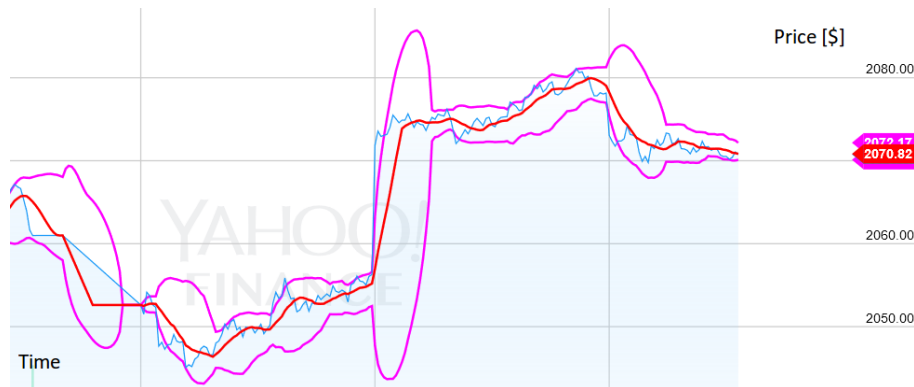
Obrázek 3.3: MACD. Modrá čára dole je MACD a žlutá je signální (finance.yahoo.com)

3.1.2 Indikátory volatility

Bollingerova pásma (Bollinger Bands)

Bollingerova pásma (BB) je indikátor založený na volatilitě. Tvoří horní a spodní pásmo kolem klouzavého průměru. Je vytvořen tak, že se přičtou nebo odečtou standardní odchylky od průměru. Protože standardní odchylka je měřítkem volatility, tak se BB rozšiřují, když dochází k nárůstu objemu na trhu. Naopak při klidnějším trhu jsou BB užší.

BB lze taky využít jako signál k nákupu pokud aktuální cena protne horní křivku. Signál k prodeji je dán pokud dojde k protnutí aktuální ceny a spodní křivky BB.



Obrázek 3.4: Bollinger Bands (fialová) 20 dní a klouzavý průměr (červená) 10 dní (finance.yahoo.com)

Average True Range

Dalším důležitým indikátorem, který měří volatilitu je ATR (Average True Range). Je vytvořen se zaměřením na krátkodobější obchody. ATR lze využít k umístování stop-lossů. Například lze stop-loss umístit 1 ATR pod vstupní cenu. Výhodou je, že stop-loss je závislý na stavu trhu a nedochází tak často k předběžnému opuštění pozice.

3.1.3 Oscilátory

Dalším důležitým typem indikátoru jsou oscilátory. Ty ukazují, zda se dá čekat obrat trendu. Nejznámější jsou RSI, Stochastic a Momentum.

RSI (Relative Strength Index)

Index relativní síly měří vnitřní sílu akcie a zobrazuje se obvykle v grafu. Hodnota se pohybuje 0 do 100.

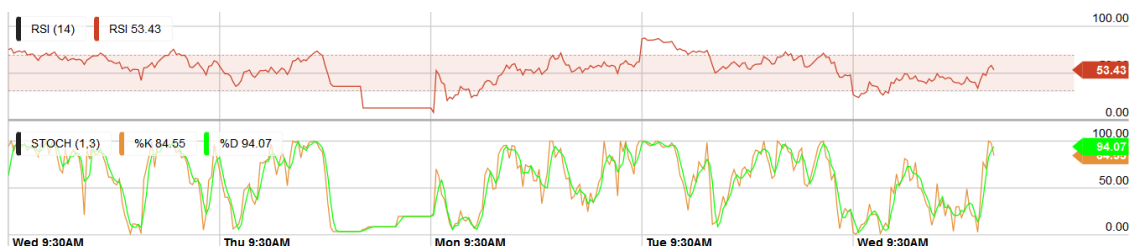
$$RSI_t(n) = 100 - \frac{100}{1 + RS_t(n)} \quad (3.3)$$

$$RS_t(n) = \frac{\text{součet kladných cenových změn za období } n}{\text{součet negativních cenových změn za období } n} \quad (3.4)$$

Jedná se o jeden z nejpoblárnějších indikátoru. Často se používá 14, 9 a 25 denní RSI. Podle základních pravidel vysílá RSI prodejní signál, pokud je pod hranicí 70 a nákupní signál nad hranicí 30. Indikuje překoupenost nebo přeprodanost trhu.

Stochastic

Křivky stochastic se pohybují v rozpětí 0% až 100%. Křivky jsou dvě a nazývají se K a D. Nejpopulárnějším nastavením je varianta 5 dní (hodinový, minutový). Tento indikátor patří k velmi volatilním. Při slabém trendu vykazuje největší spolehlivost signálu, naopak při silném trendu může být zavádějící. V porovnání s RSI je vhodnější pro méně volatilní trhy. Výhodou je, že dobře signalizuje překoupení a předprodání trhu v netrendových trzích. Také dává signály o trochu dříve, než dojde k samotnému obratu trhu. Toho lze využít při swingových obchodech.



Obrázek 3.5: Porovnání Stochastic a RSI (finance.yahoo.com)

Momentum

Momentum měří, o kolik se změnila cena aktiva za dané časové období. Měří tedy zrychlení nebo zpomalení trhu. Tento indikátor také signalizuje překoupenost/přeprodanost trhu. Lze ho využít podobně jako MACD. Pokud indikátor utvoří dno, tak je to signál k nákupu a pokud vytvoří vrchol a obrátí se, tak je to signál k prodeji.

3.2 Co by měla strategie umět

Strategie určuje vhodný okamžik na vstup do pozice. Tento signál je založen na základě indikátorů popsaných v předchozí kapitole. Lze použít vhodný indikátor, který právě co nejlépe vyhovuje situaci na trhu (trendové či netrendové trhy) nebo indikátory zkombinovat. Také lze vytvořit komplexnější indikátor pomocí nějaké metody z oboru strojového učení, rozpoznávání vzorů atp.

Ještě důležitější je určit vhodný čas k opuštění pozice. Vstup do trhu probíhá na základě nějakého odhadu vývoje, ale výslednou výši ztráty nebo zisku určuje až opuštění pozice. Přestože se trh vyvíjí požadovaným směrem, může vlivem pozdního opuštění pozice dojít ke ztrátě. Obdobně taky, pokud je pozice opuštěna předčasně, není zisk tak velký, jak by mohl.

Velkou roli hraje volba nebo odvození **stop-loss** parametru. Funkcí stop-lossu je minimalizovat ztrátu obchodníka, pokud se trh vyvíjí opačným směrem, než předpokládal. Pokud je hodnota stop-lossu moc nízká, tak dojde k velké ztrátě, než projeví svůj efekt. Ovšem pokud není jeho velikost dostatečná, může na základě jeho podnětu zase docházet k častému opuštění pozice dříve, než se objeví očekávaná změna trendu.

3.3 Zhodnocení úspěšnosti algoritmu

Pokud investor vybere strategii a zvolí vhodné indikátory, tak je potřeba naprogramovanou strategii otestovat, aby bylo možné zjistit její úspěšnost. Tato část je nejdůležitější fází

vývoje obchodního algoritmu, protože pokud nejsou správně včas odhaleny parametry a úspěšnost modelu, tak může dojít k finanční ztrátě. Komplikací také je, že trh se neustále mění a strategie, která fungovala v minulosti, nemusí vůbec fungovat v současnosti. Ke kvalitně provedenému testování je potřeba mít k dispozici odpovídající historická data, nejlépe symbolu, který bude obchodován. Zpětné testování není 100% zárukou úspěchu!

Pokud je strategie optimalizována (nebo jsou použity metody strojového učení) na určité části historických dat, tak je důležité, aby byla strategie také testována na datech, které pro tyto účely ještě nebyly použity. Pokud strategie projde těmito fázemi, je potom ještě před nasazením vhodné testovat ji přímo na „živých datech“. Po důkladném vyhodnocení může být strategie nasazena na reálný trh.

Čím menší je časový rámec dané strategie (tedy vyšší frekvence obchodů), tím těžší a dražší je získat kvalitní data.

Při vyhodnocení úspěšnosti je třeba nezapomenout na zahrnutí obchodních poplatků. Například ve forexových trzích, kde jsou poplatky za obchod variabilní, můžou tvořit rozdíl mezi průměrně úspěšnou strategií a neúspěšnou. S ohledem na toto je také vhodné navrhovat strategii.

Profit and Loss – PnL

Základním výstupem ze zpětného testování jsou průběhy zisků a ztrát. Obchodník může sledovat jejich vývoj a výslednou hodnotu ztráty nebo zisku.

Drawdown

Hodnota největšího poklesu vloženého kapitálu. Pokud je maximální drawdown roven 100\$, tak to znamená, že obchodní strategie měla za dobu obchodování nejvyšší kontinuální ztrátu ve výši 100\$.

Sharpeho poměr

Metrika, která hodnotí riziko. Tuto metriku navrhl William Sharpe². Čím je hodnota Sharpeho poměru menší, tím je výdělek riskantnější. Základní pravidlo říká, že pokud je hodnota rovna 1, tak je to dobrý výsledek. Hodnota rovna 2 je velmi dobrý a 3 vynikající výsledek.

$$S = \frac{r_p}{\sigma_p} \quad (3.5)$$

Kde r_p je průměrný očekávaný zisk a σ_p je směrodatná odchylka.

Beta

Beta je metrika rizikovosti daného portfolia vůči trhu. Tuto metriku použil jako první William Sharpe, podle kterého je pojmenována předešlá metrika. Beta popisuje citlivost portfolia k pohybům trhu. Pokud je Beta rovna hodnotě 0.5, tak je citlivost relativně malá. Pokud je Beta rovna 2.0, tak to znamená, že pohyb portfolia bude na pohyby trhu reagovat přehnaně.

$$\beta = \frac{cov(Z_p, Z_m)}{\sigma_m^2} \quad (3.6)$$

²Sharpe, William F. "Mutual Fund Performance." *Journal of Business*, 1966, pp. 119-138.

Kde $cov(Z_p, Z_m)$ je kovariance mezi pohyby trhu (Z_m) a portfolia (Z_p) a σ_m je rozptyl pohybů trhu.

Další metriky

- Celkový počet obchodů za období (popř. normalizovaný počet obchodů)
- Průměrný zisk/ztráta na jednom obchodu
- Počet úspěšných/neúspěšných obchodů
- Průměrná doba držení pozice
- Celkové náklady na poplatky

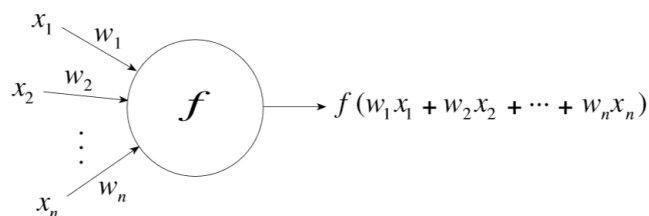
Obchodník také musí rozpoznat správný čas, kdy je strategie zastaralá a je třeba jí opustit. Lze provádět zpětné testování na aktuálních datech a sledovat jakým směrem se mění metriky dané strategie.

Kapitola 4

Neuronové sítě

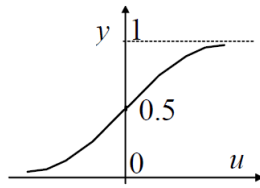
Umělé neuronové sítě jsou inspirovány biologickými neurony. Výzkum v oblasti neurovědy ve 20. století poodhalil fungování biologického mozku. Neuronové sítě se řadí mezi metody z oblasti strojového učení. Biologické neurony jsou mechanismem, který umožňuje organismům v přírodě komplexní a inteligentní chování. Organismy jsou schopny rozhodování, instinktivního chování a učení. Hlavní podstatou umělých neuronových sítí je právě schopnost učit se. Je možné pomocí nich modelovat komplexní systémy, které není právě jednoduché modelovat matematicky. Základní stavební kámen tvoří právě neuron [11].

Lidský mozek má kolem 10 až 100 biliónů neuronů, které jsou vzájemně propojeny pomocí synapsí. Těch je v mozku okolo 100 trilionů. Umělý neuron se snaží napodobit učící schopnosti lidského mozku. V následujícím textu bude uvažována pouze dopředná síť. Umělé neuronové sítě jsou právě sestaveny z těchto umělých neuronů. Sítě jsou organizovány do několika vrstev. Vrstvy jsou navzájem plně propojeny, neurony v jedné vrstvě vzájemně propojeny nejsou.

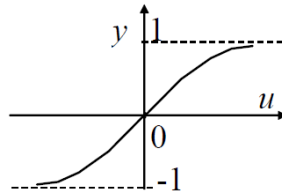


Obrázek 4.1: Umělý neuron (zdroj: [11])

Umělý **neuron** je matematická funkce, která má N vstupů (x_i až x_N) a jeden výstup. Vstup x_1 je implicitně nastaven na hodnotu jedna (tzv. bias). Každý vstup má svojí vlastní váhu (weight). Na váhované vstupy je aplikována suma a na výslednou hodnotu je aplikována tzv. aktivační funkce f . Nejčastěji hyperbolický tangens nebo sigmoida. Neuron je tedy nelineární funkce. Právě to je základ schopností neuronových sítí. K učení neuronu může být použit gradientní sestup.



Obrázek 4.2: Spojitá aktivační funkce sigmoida (zdroj: předmět SFC)



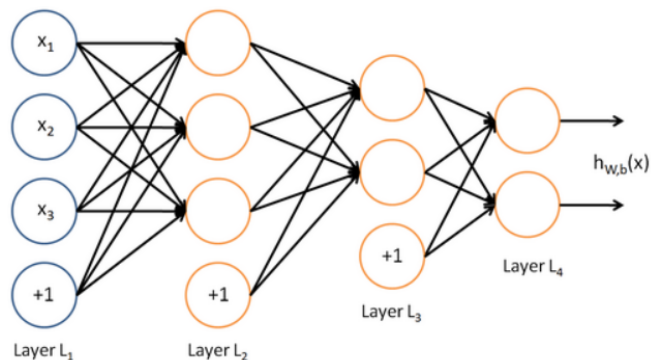
Obrázek 4.3: Hyperbolický tangens - tanh (zdroj: předmět SFC)

Neuronová síť se skládá z libovolného počtu umělých neuronů. Na neuronovou síť se dá nahlížet jako na tzv. black box. Mapuje libovolný počet N vstupu na libovolný počet M výstupů. A je to tedy funkce:

$$F : \mathbb{R}^N \mapsto \mathbb{R}^M \quad (4.1)$$

Topologie dopředné sítě (architektura) je taková, že na vstupní vrstvu o velikosti N je napojena první skrytá vrstva neuronů. Každý vstup je spojený s každým neuronem v první vrstvě. Můžou následovat další skryté vrstvy. Předchozí výstupy jsou vždy propojeny s každým neuronem v následující vrstvě. Skrytých vrstev může být libovolné množství. Poslední vrstva se nazývá výstupní. Výstupy neuronů jsou výstupy sítě.

Každému spojení je přiřazována nějaká váha. Učení sítě je právě optimalizací těchto vah. Každý neuron má ještě jako vstup hodnotu 1. Váze tohoto spojení se říká bias. Síť se dá popsat i orientovaným grafem. Nejběžnější reprezentace je na obrázku.



Obrázek 4.4: Dopředná neuronová síť. Skryté vrstvy L2, L3 a výstupní vrstva L4.

4.1 Backpropagation

Učení neuronové sítě probíhá jako učení s učitelem. Je potřeba sestavit trénovací sadu, která má dvě části a to vstupní data a k nim požadované výstupní hodnoty. Trénovací vzorky jsou tvaru (x^i, y^i) kde $x^i \in \mathbb{R}^N$, $y^i \in \mathbb{R}^M$.

Váhy neuronové sítě jsou při inicializaci nastaveny na náhodnou hodnotu z rozsahu typicky -0.5 až +0.5. Lze také použít pokročilejší heuristiky inicializace vah, které mají za cíl urychlit učení [9].

Jednou z nepoužívanějších metod k učení sítě je algoritmus **backpropagation** (zpětné šíření chyby). Mezi alternativní přístupy patří například učení pomocí PSO (Particle Swarm Optimization). Backpropagation pracuje tak, že minimalizuje chybu cenové (objektivní) funkce použitím gradientního sestupu.

Úspěšnost a průběh učení závisí na volbě cenové funkce. Cílem je najít minimum v tomto prostoru cenové funkce. Bylo by ideální, kdyby bylo možné vždy najít globální minimum, ale při použití backpropagation se stává, že učení uvízne v lokálním minimu. Chyba je propagována přes všechny neurony od výstupní vrstvy až po první skrytou vrstvu. Derivace je v backpropagation objektivní ke všem parametrům sítě. Využívá pravidla pro efektivní derivace složené funkce (chain rule)¹.

Běžná objektivní (cenová) funkce J na jednom vzorku (x, y) . Jde o takzvanou half mean squared error:

$$J(W, b; x, y) = \frac{1}{2} \|h_{W,b}(x) - y\|^2 \quad (4.2)$$

Kde $h_{W,b}(x)$ je výstup neuronové sítě pro vstupní vektor x a W jsou váhy sítě, b jsou pak biasy.

Algoritmus backpropagation je následující:

1. Dopředný průchod sítí - jsou spočítány a uloženy aktivace všech neuronů.
2. Spočítat všechny parciální derivace pro výstupní neurony.
3. Zpětný průchod - jsou spočítány parciální derivace všech neuronů v síti vždy z parciální derivace neuronů v následující vrstvě.
4. Spočítat derivace vah a biasů z aktivací předešlé vrstvy a derivací následující vrstvy.

Váhy a biasy jsou upraveny na základě následujících vztahů:

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W, b) \quad (4.3)$$

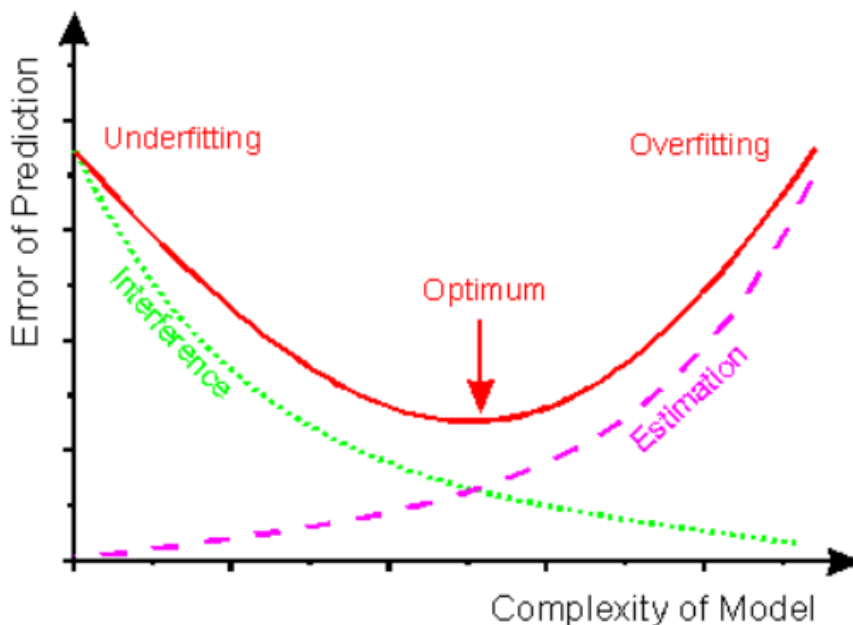
$$b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(W, b) \quad (4.4)$$

Kde $W_{ij}^{(l)}$ je j -tá váha i -tého neuronu ve vrstvě l a $b_i^{(l)}$ je bias i -tého neuronu ve vrstvě l . Hodnota α ve vztahu se nazývá koeficient učení. Udává s jakou „rychlostí“ se bude síť učit. Typické hodnoty jsou z intervalu 0.001 až 0.9. Existují i pokročilejší techniky jako adaptabilní koeficient učení.

¹Neural Networks http://ufldl.stanford.edu/wiki/index.php/Neural_Networks

4.2 Učení sítě

Datová sada vzorků je rozdělena na část validační a část trénovací. Rozdělení je vhodné pro zjištění toho, jak moc je síť schopná zobecňovat (generalizovat). Bývá hodnocena chyba na trénovací sadě a také na validační sadě. Stává se, že chyba na učící sadě se zmenšuje, ale na validační sadě vzrůstá 4.5. Tomuto jevu se říká přetrénování (overfitting).

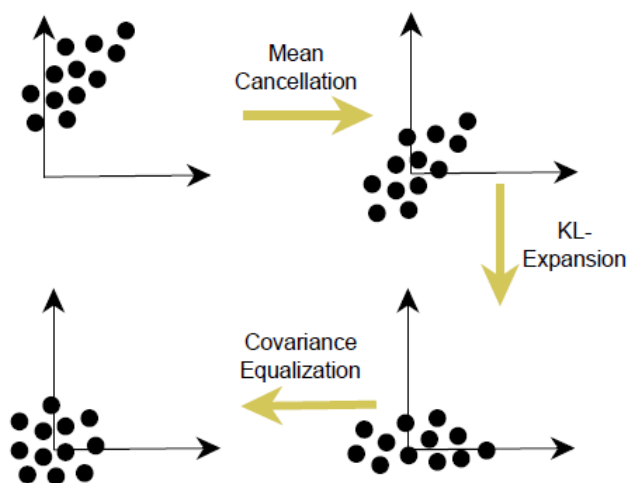


Obrázek 4.5: Chyba v průběhu učení (zdroj: www.frank-dieterle.de).

Učení na datové sadě probíhá přes několik iterací, kterým se říká epochy. Učení bývá zastaveno, pokud chyba klesne pod nějakou předem danou hodnotu, nebo pokud proběhne předem daný počet iterací. Také bývá zastaveno, pokud se chyba na validační části datové sady nemění nebo vzrůstá.

Úspěšnost učení závisí na velikosti a povaze učící datové sady. Velký význam hraje předzpracování dat. Populárnost neuronových sítí je mimo jiné zapříčiněna jejich relativní odolností vůči šumu, nicméně je vhodné z trénovacích dat odstranit chyby a nevhodné vzorky (tzv. outliers). Typicky platí, že síť reaguje na data, která ještě „neviděla“ špatnými výsledky. Problém je, že z výstupu to jde špatné poznat, protože síť neříká, jak si je výsledkem jistá.

Z předchozího vyplývá, že síť reaguje lépe, pokud je z trénovacích dat odstraněna střední hodnota. Pro tyto účely může být na vstup použita nějaká normalizační metoda. Používá se normalizace typu mean/variance nebo v rozsahu hodnot (min-max).



Obrázek 4.6: Předzpracování vstupů (zdroj: [9]).

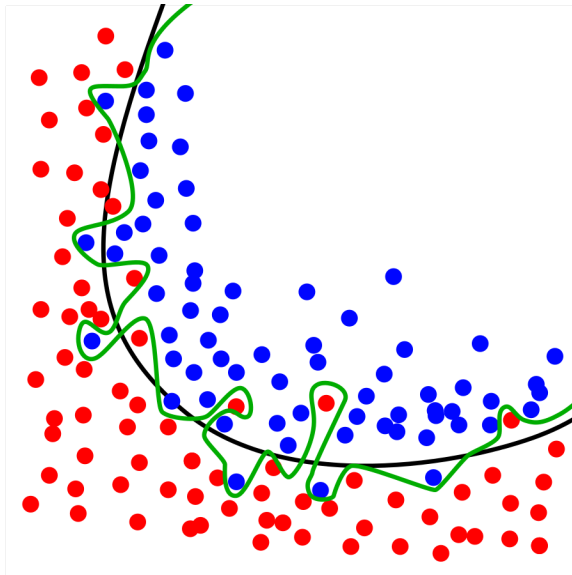
Průběh učení bývá pro přehlednost reprezentován jako graf závislosti chyby (přes celou trénovací sadu) a pořadí iterace (epochy). Do grafu je zanášena chyba na trénovací části a validační části sady.

Výsledné schopnosti neuronové sítě závisejí především na trénovací datové sadě a na volbě parametrů jako je počet skrytých vrstev, počet neuronů v každé vrstvě, volba aktivační funkce, cenové funkce a koeficientu učení. Těmto parametrům se říká hyperparametry.

4.3 Aplikace neuronových sítí

Neuronové sítě nalézají využití v mnoha disciplínách. Lze je použít jako komplexní klasifikátory (např. rozpoznávání tváří) nebo pro předpověď časových řad (předpověď počasí, předpověď vývoje na trhu). Jsou schopné také provádět kompresi dat a tak dále.

Jedním z největších problémů při aplikaci neuronových sítí je již zmíněné přetrénování (overfitting). Dá se většinou rozpoznat pomocí toho, že chyba na validační sadě začne s průběhem učení narůstat, přestože na trénovací sadě klesá. Tímto tématem se zabývá spousta publikací a existují mnoho metod, které se snaží tento neduh omezit. Platí, že čím má síť více vrstev a více neuronů, tím je k přetrénování náchylnější. Pokud už dojde k přetrénování, tak je pak vhodné použít model sítě z iterace, kde se overfitting ještě neprojevil. Na obrázku 4.7 je ukázka přetrénovaného klasifikátoru (zelená čára). Černá čára ukazuje optimálně natrénovaný klasifikátor. Ceněnou publikací v této oblasti je [9] (například LeCun a kol. "Efficient BackProp").



Obrázek 4.7: Příklad přetrénovaného klasifikátoru (zelená). Optimálně natrénovaný klasifikátor (černá).

Kapitola 5

Návrh vlastní obchodní strategie

Tato část se bude zabývat návrhem obchodního algoritmu využívajícího neuronové sítě. Návrh je postaven na principech technické analýzy. Neuronová síť bude mít za úkol předpovídat budoucí členy časové řady na základě současných a minulých hodnot. Výsledky neuronové sítě závisí především na zvolených vstupních datech, předzpracování a na nastavení jednotlivých parametrů učení a topologie.

Strategie budou vyhodnoceny a porovnány mezi sebou a vyhodnocení bude probíhat pomocí simulace obchodování na datech, které byly použité jak pro trénování, tak i na datech, které neuronová síť "nezná".

Základní myšlenka, která stojí za použitím neuronových sítí (nebo obecně strojové učení v obchodování) je, že v datech lze objevit různé opakující se vzory, které vedou k podobným výsledkům. Nejjednodušší úlohou by mohlo být určení, zda bude trend v následující časové periodě stoupající nebo klesající. To, co se má síť naučit, je čistě v režii programátora, jenže často není jednoduché identifikovat, zda se síť naučila to, co po ní požaduje, jak dobře a také existuje nebezpečí přetrénování.

5.1 Použitá obchodní data

Kvalitní obchodní data jsou jedním ze základních předpokladů k vytvoření úspěšného obchodního systému. Důležitým parametrem historických dat je jejich časové rozlišení. Čím jemnější rozlišení, tím těžší je data získat. Může jít o měsíční, týdenní nebo denní až minutová data. Pro taková data je charakteristické, že mají pro každé časové období čtyři hodnoty. Také většinou obsahují informaci o objemu transakcí (angl. volume).

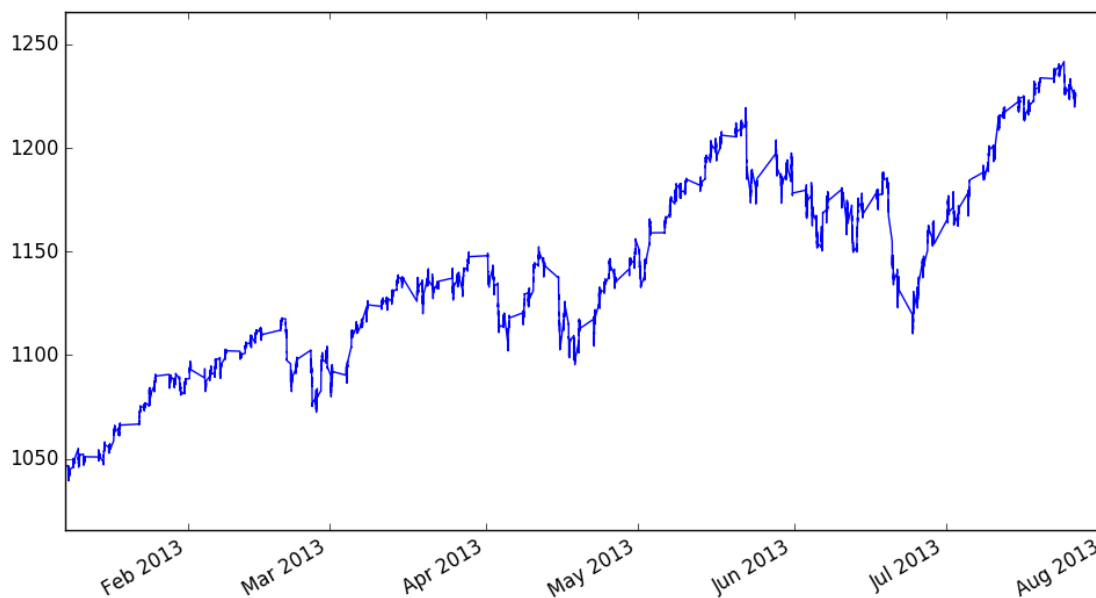
- OPEN je cena v počátku časové periody
- CLOSE je cena na konci periody
- LOW je nejnižší cena za periodu
- HIGH je nejvyšší cena za periodu

Nejhodnotnější jsou ticková data. Přístup k nim je zpravidla placený. Tick je nejmenší možná hodnota, o kterou se může cena trhu změnit. Vzhledem k tomu, že účel této práce je denní obchodování, připadají v úvahu minutová nebo ticková data.

Obecně platí, že čím je dat více, tím pro učení lépe. Při učení za účelem obchodování to ale nemusí platit vždy. Platí-li předpoklad, že se v určitém trhu vyskytují nějaké vzory,

které se využijí k předpovědi budoucích hodnot, tak nelze usuzovat, že tyto vzory budou stále stejné a časem neměnné. Všechny obchodní strategie mají omezenou dobu funkčnosti a po nějakém čase zastarávají, protože se povaha trhu změnila. Z toho vyplývá, že stará data mohou obsahovat neaktuální vzory a informace.

Price [€]



Obrázek 5.1: Použitá obchodní data symbolu EMD.

Symbol EMD

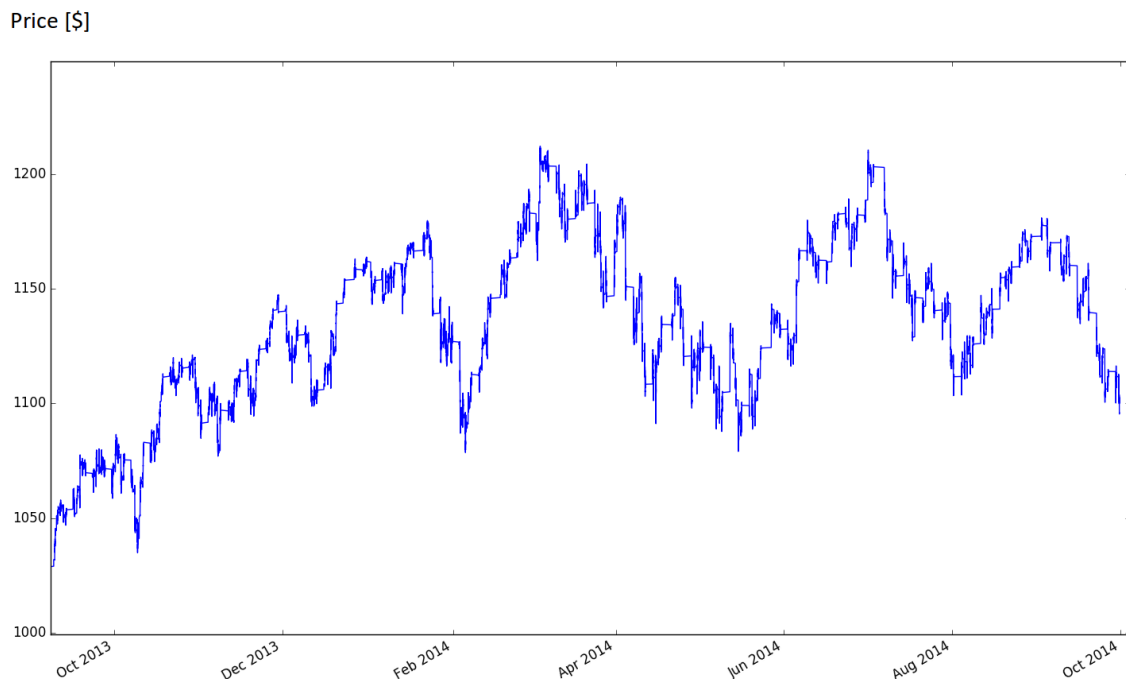
K učení a testování jsou použita minutová data, která mají symbol EMD. EMD znamená E-Mini futures kontrakty (viz kapitola 2.1.3) na S&P MidCap 400 Index.

Data obsahují 56978 vzorků a jsou z období 2.1.2013 až 26.7.2013, tedy zhruba půl roku a mají celkově rostoucí trend, i když se v některých obdobích vyskytují znatelné propady ceny. Při koupi jedné transakce na počátku období a při prodeji na konci by byl generován zisk 19 000\$, protože rozdíl mezi počáteční cenou a konečnou cenou je 190\$ a pákový efekt při obchodování tohoto produktu je 100 násobný.

Zdroj dat je minutový. Výhodou E-Mini futures je, že se obchodují téměř 24 hodin denně a v současné době jde o velmi populární trh. Obchoduje se na burze GLOBEX. Poplatky jsou 2.5\$ za transakci.

Parametry	EMD	TF
Vzorků	56978	112302
Od	2.1.2013	4.9.2013
Do	26.7.2013	30.9.2014
Volatilita (prům. za 100 min)	1.46	1.82
Pákový efekt	100x	100x
Procentuální změna	+18 %	+9.6 %
Poplatky za transakci	2.5\$	2.5\$

Tabulka 5.1: Parametry jednotlivých použitých datových sad.



Obrázek 5.2: Použitá obchodní data symbolu TF.

Symbol TF

Další použitá data mají symbol TF. Jde o futures založené na E-Mini Russell 2000. Zdroj dat je tickový. Data byla navzorkována na minutové rozlišení, protože pro ticková data v této práci není využití.

Při zakoupení jedné položky (za 1007\$) na začátku období a prodeji na konci období (za 1096\$) by byl generován zisk 8900\$. Data jsou za období od 4.9.2013 po 30.9.2014. Data nemají jasný trend jako symbol TF. Obsahují jak velké propady, tak prudké nárůsty hodnoty. Pákový efekt je stejně jako u předchozích dat roven 100.

TF se obchoduje na burze NYBOT. Tento trh je obchodován téměř 24 hodin denně s přestávkou mezi 18:00 a 20:00 ve všední dny. Poplatky jsou stejně jako u symbolu EMD 2.5\$ za transakci.

Poplatky

U popsaných dat je zmíněno, že poplatky za obchodování činí 2.5\$. Tato hodnota se může mírně lišit broker od brokera, ale dá se říct, že velikost poplatků je právě jedna z výhod obchodování E-Mini Futures, protože nejsou moc velké a jsou fixní. Platí se za provedený kontrakt. To znamená, že poplatek za jeden obchod je součtem poplatku za vstup do pozice a poplatku za výstup z pozice.

Poplatek brokerovi je většinou 0.5\$ až 1.0\$, tato cena se také odvíjí od počtu provedených transakcí za určité období. Pokud obchodník provádí velké množství transakcí, může dojít ke snížení poplatku, ale ještě je potřeba uhradit poplatky burze a tak podobně, výsledná cena poplatku je tedy standardně kolem 2.5\$.

5.2 Testování obchodní strategie

Pro robustní testování obchodních strategií existují pro Python kvalitní knihovny, které se snaží s určitou úrovní věrnosti simulovat reálné obchodování. Dělí se na dva základní typy. Ty lepší jsou tzv. event-driven. Tento typ implementace snižuje riziko toho, že by obchodní strategie získala nějaká data z budoucnosti. Strategie je definována jako sada pravidel, které jsou v každém novém časovém intervalu vyhodnocovány na aktuálních datech. Podle vyhodnocení je pak proveden vstup do pozice nebo její opuštění. Pravidla mohou být zkonstruována libovolně. V tomto případě využívají predikci z neuronové sítě, která je založena pouze na předchozích datech.

Při vstupu do pozice v reálném trhu není jisté za jakou cenou bude tato transakce provedena, protože hodnoty poptávky a nabídky se neustále mění. Aby nedošlo k provedení příkazu za jinou cenu, než je uživatel ochotný nabídnout, tak existují tzv. limitní příkazy. Součástí příkazu je hraniční hodnota, pro kterou může být proveden. Příkazy lze zadávat také bezlimitně, takže jsou provedeny za libovolnou aktuální cenu na trhu.

Při zpětném testování je to simulováno většinou tak, že rozhodování nad vstupem nebo výstupem z pozice se provádí nad aktuální hodnotou CLOSE. Ale cena, za kterou se transakce pak provede, je hodnota OPEN z následující časové periody. Zjednodušeně to může být implementováno i tak, že cena při provedení bude aktuální CLOSE.

Vlastní implementace zpětného testování

Tato práce nakonec využívá vlastní implementaci backtestingu, protože vyzkoušené knihovny (Backtrader¹) byly několikanásobně pomalejší a vlastní implementace také nabízí větší flexibilitu týkající se shromažďování dat apod.

Simulace využívá minutová data. Tyto data jsou postupně procházena v cyklu. Strategie má přístup k OPEN, LOW, CLOSE, HIGH hodnotám. Na základě těchto dat vyhodnotí další akci. Strategie může vstoupit buď do krátké nebo dlouhé pozice. Také je možné stanovit hodnotu stop-loss, při které dojde k opuštění dané pozice. Četnost jeho použití bude vyhodnocena na výstupu ze simulace.

Strategie může současně držet pouze jednu pozici. Finančně je neomezená. Začíná s částkou 0\$ a při vstupu do pozice nakoupí vždy 100 jednotek. Tím se simuluje pákový efekt při obchodování E-Mini Futures symbolu EMD a TF.

Protože cílem je denní obchodování, je kontrolováno, aby pozice nebyla držena mimo obchodní hodiny nebo v časových úsecích, kdy data nenavazují a chybí delší období. Je to

¹Backtrader <http://www.backtrader.com/>

z toho důvodu, že hodnota trhu se může skokově změnit a to může negativně ovlivnit stav účtu a také schopnost predikce neuronové sítě.

Vyhodnocení úspěšnosti obchodní strategie

- Konečný stav účtu
- Počet obchodních transakcí
- Průměrná doba držení pozice
- Počet ziskových obchodů
- Počet ztrátových obchodů
- Počet stop-lossů
- Průměrná ziskový obchod
- Průměrná ztrátový obchod
- Sharpeho poměr
- Beta
- Maximální drawdown

Shrnutí základních vlastností

- Poplatky za obchodování jsou odečteny dodatečně
- Strategie může vstoupit do krátké a dlouhé pozice
- Současně může držet pouze jednu pozici
- Strategie není finančně omezena
- Pákový efekt 100x
- Variabilní stop-loss
- Pozice je ukončena před koncem dne

5.3 Obchodování pomocí neuronové sítě

Jak už bylo naznačeno, neuronová síť bude mít za úkol na základě několika vzorků vstupní časové řady předpovídat její následující vzorek. Jde o klasickou regresní úlohu, ke které může být použita jednoduchá dopředná síť. Neuronová síť může být použita také jako klasifikátor. Trénovací vektory lze rozřítit do skupin, které určují, zda je na základě vstupního vektoru vhodné vstoupit do pozice nebo z ní vystoupit.

Neuronové sítě byly pro podobnou úlohu využity už mnohokrát. Například CW Tan [13] nebo [1]. Normalizací dat a návrhem sítě pro tuto úlohu se zabývá Kaastra [6].

Základním vstupem neuronové sítě jsou předzpracované členy časové řady historických a aktuálních hodnot. Otázka je, kolik členů je potřeba k co nejpřesnější předpovědi a zda může i nadbytečný počet (redundantních vstupů) uškodit přesnosti predikce.

Technická analýza může na základě technických indikátorů poskytnout vodítka k tomu, kdy vstoupit nebo vystoupit z pozice. Z tohoto pohledu by mohlo přidání technických indikátorů na vstup neuronové sítě pomoci zlepšit kvalitu předpovědi.

Další volbou, kterou je třeba učinit, je architektura sítě a parametry učení. Existují určité heuristiky, kolik by měla mít síť skrytých vrstev a kolik neuronů. Jak lze vidět, tak parametrů, které mohou ovlivnit výslednou schopnost regrese neuronové sítě, je opravdu mnoho. Je možné volit buď systematicky, ale to počet kombinací nedovoluje, nebo na základě intuice. Další možností je využít pro optimalizaci evoluční algoritmus. Ten by mohl najít vhodnou kombinaci hyperparametrů. Hodnotící funkcí by byla přesnost predikce neuronové sítě.

5.3.1 Předzpracování vstupních dat

Jedna z nejdůležitějších operací ve strojovém učení je předzpracování dat, konkrétně jejich normalizace. Pokud nejsou data normalizovaná, tak se síť zpravidla nic nenaučí. Protože cílem je naučit se vzory ve vstupních datech, je potřeba od dat alespoň odečíst jejich průměr. Předpoklad je, že vzory (patterns) na trhu budou fungovat podobně a to nezávisle na jejich hladině. Jestliže bude průměrná cena v nějakém období například 1000\$ nebo v jiném období 800\$, tak by to nemělo zásadně měnit relativní závislosti mezi jednotlivými vzorky. Je možné také použít vzájemné diference hodnot, to by měla být teoreticky vhodnější reprezentace pro učení vzorů, ale tato metoda při experimentování nefungovala dobře.

Vstup lze jednoduše přeškálovat do požadovaného intervalu nebo aplikovat tzv. mean-variance normalizaci, která zajistí, že data mají nulový průměr a jednotkovou varianci. Vybrané metody jsou popsány níže.

Normalizace pomocí SMA

Místo normalizace pomocí SMA (jednoduchého klouzavého průměru) je možné také využít exponenciální klouzavý průměr (EMA). Rozhodující pro volbu délky období klouzavého průměru je, na jaký časový rozsah by měl být obchod cílen. Délka tohoto období výrazně ovlivňuje citlivost k aktuálním hodnotám. Při kratším období bude síť citlivější na menší změny, tím ale může přijít o citlivost na dlouhodobější trendy.

Data jsou normalizována tak, že je od nich odečtena aktuální hodnota SMA. Následně jsou ještě vydělena směrodatnou odchylkou (směrodatná odchylka dat je pak jednotková).

Výpočet normalizované hodnoty vstupu:

$$n_{t-i} = \frac{x_{t-i} - m_t}{v_t}, \text{ pro } i = 0, \dots, n \quad (5.1)$$

Kde n_{t-i} je normalizovaná hodnota ceny pro x_{t-i} . m_t je hodnota klouzavého průměru pro čas t a v_t je směrodatná odchylka pro čas t . Proměnná n představuje požadovaný počet normalizovaných hodnot.

Normalizace pomocí logaritmu

Podle [12] je vhodné normalizovat obchodní data pomocí logaritmické funkce. Výhodou oproti předchozí normalizaci je, že není nutné volit časové období pro výpočet SMA. Postup

je takový, že se na vstupní vektor aplikuje logaritmická funkce. Následně jsou hodnoty transformovány buď tak, že je od nich odečten jejich průměr (zlogaritmovaný) nebo jsou přeškálovány do vhodného intervalu (například -1 až 1).

Přeškálování může být provedeno na základě nejmenší a největší logaritmické hodnoty zpracovávaného vektoru nebo na základě nejmenší a největší hodnoty za delší časový období (timeframe). Podobně jako u normalizace pomocí SMA.

Varianta této metody je zlogaritmování vzájemných diferencí vstupních dat a aplikování předchozích metod. Použití vzájemných diferencí by mělo silněji zohledňovat vzájemné relativní pohyby mezi jednotlivými vzorky vstupního vektoru, ale jak už bylo řečeno, při experimentech nebyla tato metoda úspěšná.

Transformace na uniformní distribuci

Tato metoda může být použita i na nenormalizovaná data. Přestože uniformní distribuce neodpovídá distribuci vstupních dat, tak by mohla podle [7] zlepšit predikční schopnosti neuronové sítě. Tato metoda je implementována jinak než v citované práci.

Mezi jednoduché metody, jak tohoto docílit patří nalezení percentilového pořadí (percentile rank)² pro každý vzorek ze vstupního vektoru, který má být zpracován. Tyto ranky jsou uloženy v novém vektoru (na pozici stejnou, jako měl odpovídající vzorek ve vstupním vektoru).

Pak už zbývá pouze tento vektor přeškálovat do vhodného intervalu (např. od -1 po +1).

Odvození požadovaného výstupu

Odvození požadovaného výstupu (target) může být problematické. Rozsah použitelných hodnot závisí na definičním oboru aktivační funkce ve výstupní vrstvě neuronové sítě. Pro logistickou funkci je to (0,1), pro hyperbolický tangens (-1,1) a pro lineární funkci teoreticky $(-\infty, +\infty)$. Při použití lineární aktivační funkce na výstupu není tedy třeba omezit rozsah požadovaných hodnot, ale ukázalo se, že je pro učení vhodnější, když jsou hodnoty v rozsahu blízkém (-1,1).

Otázkou je, co by vůbec měl výstup reprezentovat. To je zcela v rukou programátora. Vhodné je, aby byl výstup nějak odvozen od použitých hodnot tak, aby byla možná zpětná interpretace výstupu při reálném použití. Může to být třeba podíl predikované hodnoty a aktuální hodnoty ceny nebo jejich rozdíl. Také může být odvozen od průměrné hodnoty vstupních dat. Tato výstupní hodnota může být pak ještě nějak normalizována. To může ale narušit závislost na vstupních hodnotách a tím znemožnit učení. Problém také je, že normalizace komplikuje právě zpětnou interpretaci výstupu.

Nakonec je pro požadovaný výstup použit tento vztah:

$$p_{t+1} = \frac{x_{t+1} - m_t}{5v_t} \quad (5.2)$$

Kde p_{t+1} je požadovaný výstup (target) neuronové sítě pro trénovací vektor, hodnota x_{t+1} je cena trhu v čase $t + 1$. Od té je odečten klouzavý průměr m_t spočítaný původně pro normalizaci vstupního vektoru (rovnice 5.1). v_t je směrodatná odchylka v čase t , který byla také počítána z daného počtu historických hodnot za účelem normalizace vstupního vektoru.

²Percentile rank: <http://www.regentsprep.org/regents/math/algebra/ad6/quartiles.htm>

Počet historických hodnot, z kterých je počítána hodnota klouzavého průměru a směrodatné odchylky, se nazývá timeframe. Tato hodnota byla zvolena jako 35. Pokud je hodnota větší, tak se ztrácí citlivost k aktuálním datům a pokud je menší, tak se naopak ztrácí kontext a citlivost k dlouhodobějším trendům.

Hodnota 5 ve vzorci byla ma základě několika experimentů zvolena tak, aby výstupní hodnoty odpovídaly co nejlépe oboru hodnot výstupní vrstvy neuronové sítě. Několik málo hodnot je větších než 1 a menších než -1.

Interpretace výstupu je pak při predikci následující:

$$y_{t+1} = 5v_t p_{t+1} + m_t \quad (5.3)$$

Kde p_{t+1} je výstup neuronové sítě (predikce) a y_{t+1} je přímo predikovaná cena.

Obecně může být pro predikci využita jiná hodnota než $t + 1$, jako třeba $t + 10$ nebo průměrná hodnota z rozsahu $t + 1$ až $t + 10$. Hodnota $t + 1$ bude pravděpodobně velmi blízko hodnotě v čase t , ale predikce vzdálenější hodnoty bude pro neuronovou síť obtížnější.

Topologie sítě a její trénování

Topologie sítě je také jedním z parametrů, které ovlivňují přesnost predikce. Obecně platí, že síť o dvou skrytých vrstvách je schopná naučit se aproximovat jakoukoliv funkci s libovolně malou chybou³. Použití více vrstev má v dnešní době velký význam především v oborech zpracování obrazu jako jsou konvoluční neuronové sítě a hluboké sítě.

Použití nadbytečného počtu skrytých vrstev a neuronů v nich může mít neblahé důsledky na schopnosti neuronové sítě. Síť pak může být náchylnější k přetrénování a trénování trvá celkově déle. Oproti tomu malý počet neuronů nemusí dostačovat.

Každá skrytá vrstva může mít na výstupu jinou aktivační funkci. Standardně se používá lineární funkce, logistická funkce nebo hyperbolický tangens. Nevypadá to, že by volba aktivační funkce ve skrytých vrstvách měla nějaký zásadní vliv na schopnosti sítě. Na vstupu a výstupu je vhodné volit aktivační funkce s ohledem na rozsah hodnot.

5.4 Implementace

Neuronové sítě jsou implementovány pomocí hojně používané knihovny Keras⁴ v jazyce Python 2.7. Tato knihovna byla vybrána, protože obsahuje spoustu rozšíření a je dobře optimalizovaná. Také je dobře dokumentována a má velkou komunitu aktivních uživatelů. Keras využívá Theano⁵, takže výpočty mohou být počítány paralelně na GPU, což je při učení sítě mnohem rychlejší než na CPU.

Mezi výhody Kerasu patří modularita. Vytvoření sítě je velmi jednoduché, stejně jako změna různých parametrů topologie a učení. Při tvorbě sítě využívá sekvenční model, jednotlivé vrstvy s danými parametry jsou jednoduše a přehledně přidávány do sítě.

Dále jsou využívány klasické knihovny v Pythonu jako je NumPy, SciPy a matplotlib. Zajímavá je knihovna Pandas⁶. Ta je určena pro statistické zpracování objemných dat. Cennou funkcí je například analýza dat na chybějící záznamy a možnost jejich aproximace pomocí známých hodnot.

³Andoni et al., 2014 "Learning Polynomials with Neural Networks"(2014)

⁴Keras <http://keras.io/>

⁵Theano <http://deeplearning.net/software/theano/>

⁶Pandas <http://pandas.pydata.org/>

Kapitola 6

Vyhodnocení regresní obchodní strategie

V této části bude na základě použitých dat, které byly specifikované v předchozí kapitole, vyhodnocena obchodní strategie. Datová sada symbolu EMD se bude označovat jednoduše EMD a stejně tak datová sada symbolu TF. Proces simulace obchodování je popsán v kapitole 5.2.

Grafy pro stav účtu v průběhu trénování jsou odděleny vertikální čarou (např. graf 6.5). Levá část od dělicí vertikální čáry reprezentuje trénovací část. Pravá část představuje testovací část datové sady (nebylo na ní trénováno). Výsledky testování na obou částech jsou zaneseny do tabulky. Tabulka obsahuje výsledky pro obě části, aby bylo možné výsledky na těchto částech navzájem porovnat. Lze z toho usoudit něco o tom, jak je schopna síť generalizovat.

6.1 Referenční obchodní strategie

Tato jednoduchá obchodní strategie využívá pouze SMA (simple moving average - jednoduchý klouzavý průměr). Strategie interpretuje výchylky od průměrné ceny jako signály. Pokud se cena dostane nad SMA, tak je to signál k vstupu do dlouhé pozice. Pokud se naopak cena dostane pod SMA, tak je to signál k výstupu z dlouhé pozice (vstupu do krátké).

Z porovnání s neuronovými sítěmi nepůjde přímo vyvodit nějaké konkrétní závěry, ale bude zajímavé sledovat, zda strategie prodělávají v podobných místech, jak se bude lišit zisk a jiné parametry.

Období pro výpočet SMA je jediný parametr, kterým je možné tuto strategii ovlivňovat. Období pro SMA bylo nastavena na 20, 50 a 100 vzorků (minut). Vstup do dlouhé pozice pro období 50 minut:

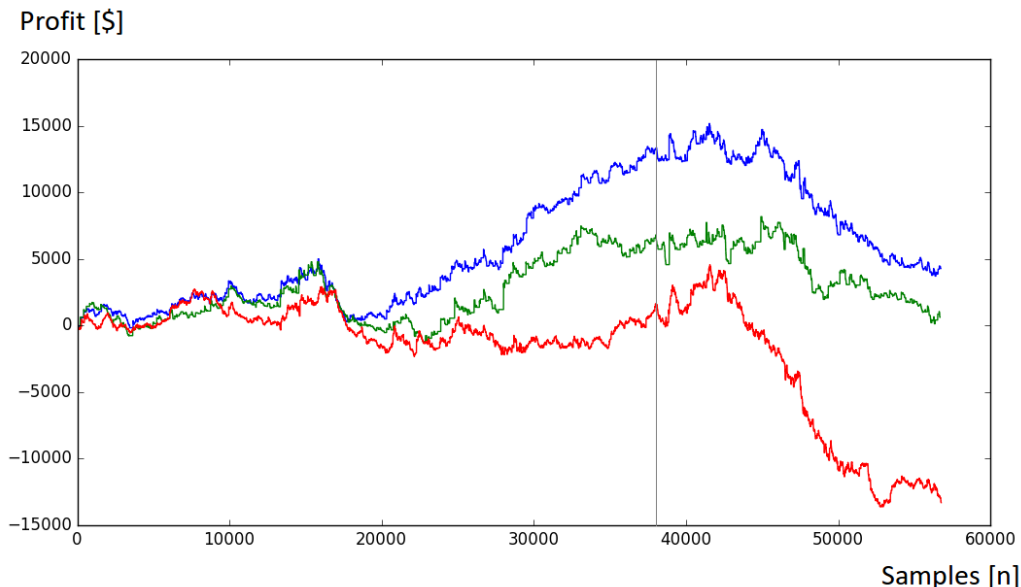
$$x_t \geq \text{SMA}(50) \tag{6.1}$$

Vstup do krátké pozice pro období 50 minut:

$$x_t \leq \text{SMA}(50) \tag{6.2}$$

Kde x_t je aktuální cena na trhu. Stop-loss je nastaven fixně tak, že je aktivován, pokud je aktuální pozice ve ztrátě 300\$.

Obě datové sady jsou uměle rozděleny na dvě části (viz vertikální čára na grafech 6.1 a 6.2). Nazývají se **trénovací a testová**, protože to odpovídá použití dat v neuronové síti. Aby byly výsledky porovnatelné s výsledky testování na neuronových sítích, jsou tato data také tak rozdělena (2/3 dat je označeno jako trénovací a 1/3 jako testovací).

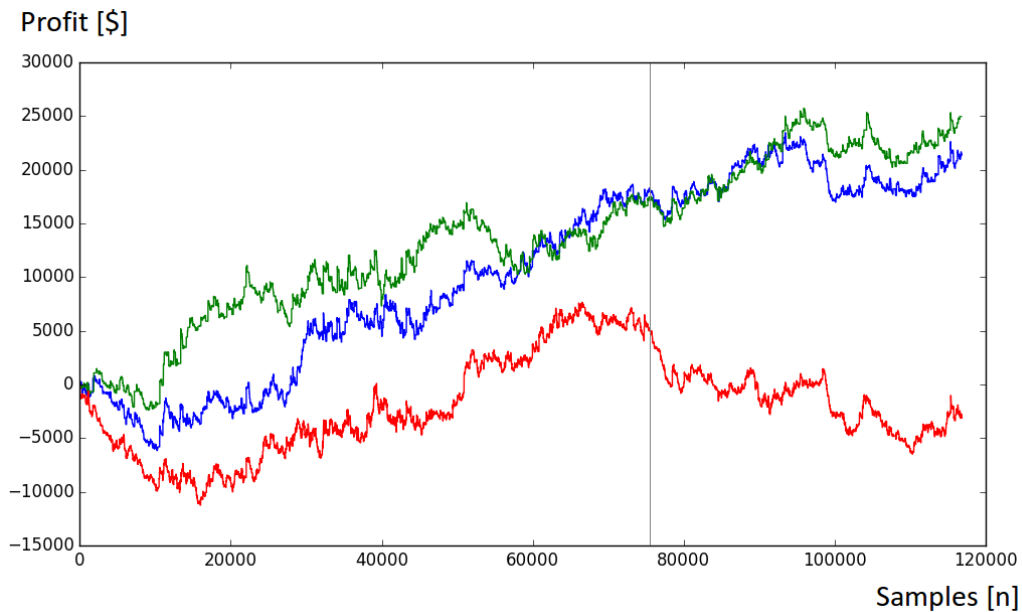


Obrázek 6.1: Průběh obchodování referenční strategie na datech EMD. SMA za období 20 minut (červená), 50 (modrá) a 100 (zelená).

Z výsledků lze vidět, že v obou případech byl počet ztrátových obchodů hodně vysoký (cca 80%). Počet stop-lossů byl velmi malý, takže mohl být nastaven na menší ztrátu. Jeho hodnota je fixně nastavena na ztrátu ve výši 300\$. Existují lepší metody jak stop-loss nastavit dle situace na trhu, tato strategie by z nich mohla benefitovat. Také se ukázalo, že pokud bylo období SMA větší, počet obchodů byl menší.

Strategie si vedla na datech TF o dost lépe než na datech EMD. V tabulce 6.1 a 6.2 jsou podrobné výsledky pro nejlepší strategii na TF a pro nejlepší na EMD. Strategie na TF se SMA, který měl období nastaveno na 100, byla v obou částech zisková. Nejlepší strategie na EMD (období 50) byla v testové části ve ztrátě -13542\$. Zajímavé je, jak se strategie na EMD stala ze ziskové (v první části) ztrátovou (v testovací části). To vypovídá o tom, že strategie, může být sice úspěšná, ale stačí, aby došlo k nějaké změně v povaze trhu a najednou se stane ztrátovou.

Také lze pozorovat, že u TF byly průměrné ziskové a ztrátové obchody větší než u EMD. To je pravděpodobně způsobeno tím, že data TF jsou volatilnější než EMD (viz tabulka 5.1).



Obrázek 6.2: Průběh obchodování referenční strategie na datech TF. SMA za období 20 minut (červená), 50 (modrá) a 100 (zelená).

Metriky	Trénovací část	Testovací část
Hrubý zisk	10700\$	-7340\$
Zisk po odečtení poplatků	1893\$	-13542\$
Počet obchodů	1760	1240
Průměrná doba v pozici	16 min	14 min
Počet ziskových	401	256
Počet ztrátových	1360	984
Počet stop-lossů	1	26
Průměrný ziskový obchod	165.58\$	200.85\$
Průměrný ztrátový obchod	-40.95\$	59.71\$
Sharpeho poměr	0.67	0.09
Max drawdown	4730\$	11450\$
Beta	0.35	0.08

Tabulka 6.1: Tabulka výsledků pro referenční obchodní strategii na datech EMD pro SMA 50.

Metriky	Trénovací část	Testovací část
Hrubý zisk	15180\$	9460\$
Zisk po odečtení poplatků	2158\$	3233\$
Počet obchodů	2604	1247
Průměrná doba v pozici	22 min	23 min
Počet ziskových	496	237
Počet ztrátových	2108	1010
Počet stop-lossů	1	26
Průměrný ziskový obchod	351.22\$	306.49\$
Průměrný ztrátový obchod	-75.44\$	-62.55\$
Sharpeho poměr	0.26	0.33
Max drawdown	6710\$	5520\$
Beta	0.06	0.07

Tabulka 6.2: Tabulka výsledků pro referenční obchodní strategii na datech TF pro SMA 100.

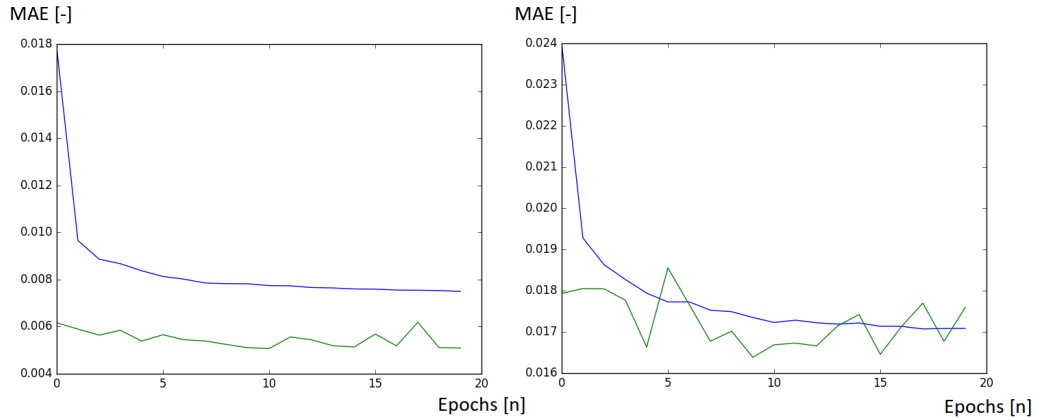
6.2 Regresní neuronová síť

Tento jednoduchý model je založen na použití neuronové sítě ve formě nelineární regrese. Neuronová síť na základě historických hodnot predikuje hodnotu trhu v následujícím časovém okamžiku. V případě minutových dat je to predikce hodnoty CLOSE v následující minutové periodě. Vstupní data jsou normalizována pomocí SMA, tento postup je popsán v kapitole 5.3.1, stejně jako odvození výstupu v kapitole 5.3.1.

Pro EMD probíhá učení zhruba na 30000 vektorech o délce 15 vzorků s jednou výstupní hodnotou. Přestože trénovací část dat EMD obsahuje více než 37500 vzorků, trénovacích vektorů je pouze 30000, protože jsou vyřazeny vektory, v nichž nejsou kontinuální data. To se může stát vždy mezi koncem obchodování a začátkem obchodování následujícího dne.

Pro TF je trénovacích vektorů více než 56000, ale vstupní trénovací část má více než 73000 vzorků. Důvod pro menší počet trénovacích vektorů je stejný jako v předchozím případě.

Validační část je volena náhodně a tvoří ji 10% ze vstupních dat. Cenová funkce je MAE (mean absolute error). Graf chybové funkce na trénovací a validační sadě je v obrázku 6.3.



Obrázek 6.3: Pokles chyby u trénování u EMD a TF (zleva). Zelená křivka je chyba na validační sadě, modrá na trénovací.

Úspěšnost učení lze graficky ilustrovat pomocí regresního grafu, který ukazuje závislost požadovaných výstupních hodnot (target values) na predikovaných hodnotách po natrénování sítě (viz obrázek 6.4). Topologie této sítě je popsána v tabulce 6.3, byla zvolena na základě několika experimentů.

Vrstva	Aktivační funkce	Neuronů
Vstupní vrstva	lineární	17
1. Skrytá vrstva	tanh	15
2. Skrytá vrstva	tanh	11
Výstupní vrstva	lineární	1

Tabulka 6.3: Topologie regresní sítě.

Obchodní strategie

Strategie se rozhoduje podle rozdílu aktuální hodnoty CLOSE a predikované hodnoty CLOSE. Pokud je predikovaná hodnota větší než aktuální hodnota, tak je očekáván nárůst v ceně trhu. Tudíž dojde k vstupu do dlouhé pozice a výstupu z krátké pozice, pokud se v ní strategie právě nachází. Obdobně, pokud je predikovaná hodnota menší než aktuální hodnota, tak dojde ke vstupu do krátké pozice a výstupu z dlouhé pozice. Tato pravidla lze parametrizovat tak, že rozdíl mezi aktuální hodnotou a predikovanou hodnotou musí být větší než nějaký parametr, aby došlo k danému rozhodnutí. Velikost tohoto parametru by měla být volena tak, aby byl omezen šum v predikci, který by mohl vést k nechtěnému rozhodnutí a tak, aby nebyly zbytečně filtrovány korektní predikce.

Zvolený timeframe pro SMA, kterým jsou normalizovány vstupní hodnoty NN je 35 minut. Následující prahové hodnoty byly určeny na základě experimentů (-0.3 a +0.3).

- Pravidlo pro vstup do dlouhé pozice:

$$(x_{pred} - x_0) > 0.3 \tag{6.3}$$

- Pravidlo pro výstup z dlouhé poice:

$$(x_{pred} - x_0) < -0.3 \tag{6.4}$$

- Pravidlo pro vstup do krátké pozice:

$$(x_{pred} - x_0) < -0.3 \quad (6.5)$$

- Pravidlo pro výstup z krátké poice:

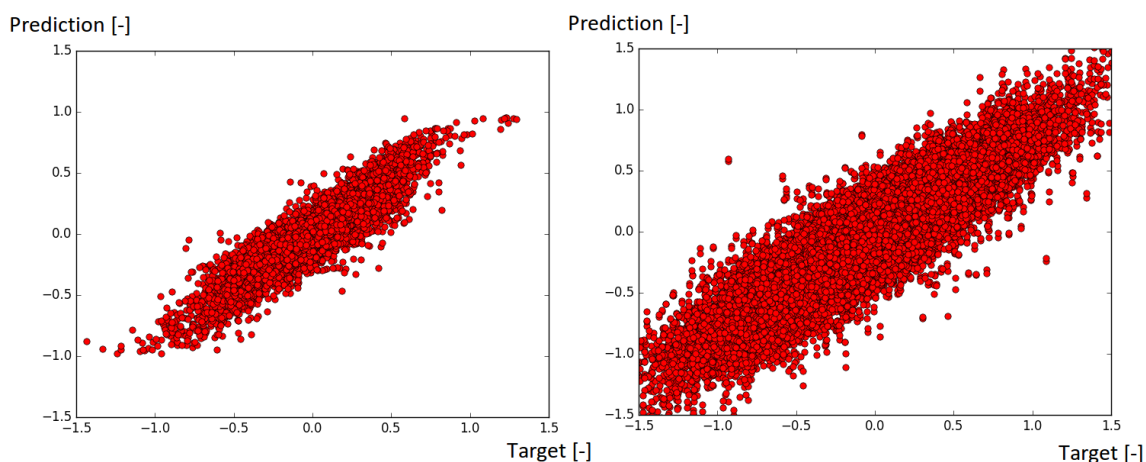
$$(x_{pred} - x_0) > 0.3 \quad (6.6)$$

Současná cena na trhu je x_t a x_{pred} je predikovaná hodnota pomocí neuronové sítě. Výpočet predikované hodnoty trhu je uveden v kapitole 5.3.1.

Stop-loss je nastaven **fixně** tak, aby byl aktivován pokud je ztráta na aktuální pozici větší než 200\$ (posun trhu o dva body nepříznivým směrem).

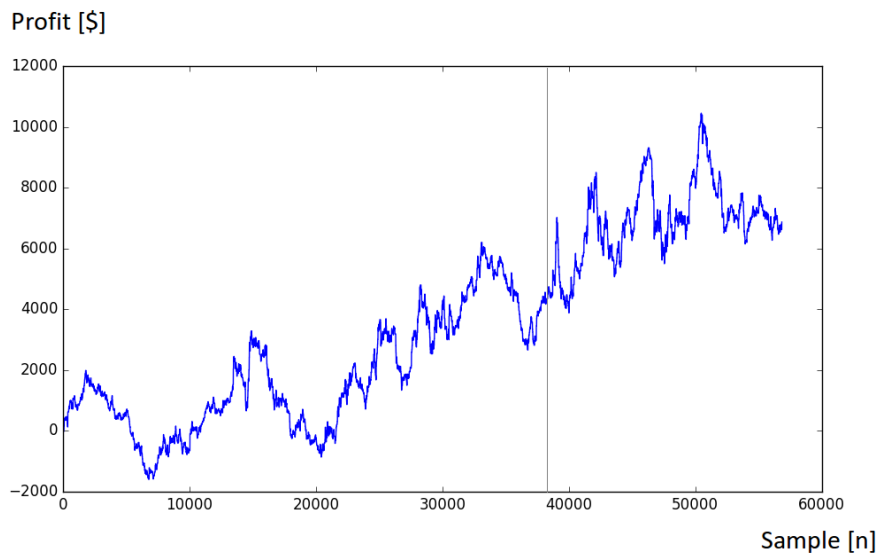
6.2.1 Výsledky pro učení bez indikátorů

Sítě byly učeny na 2/3 dostupných dat a na zbylé třetině byly testovány, aby se ukázalo, zda je síť schopná generalizovat. Na vývoji stavu účtu v čase lze vidět dělicí čára (graf 6.5). Oblast vlevo od této čáry odpovídá testování na části dat, na které byla síť učena. Oblast vpravo odpovídá části dat, na které síť učena nebyla.

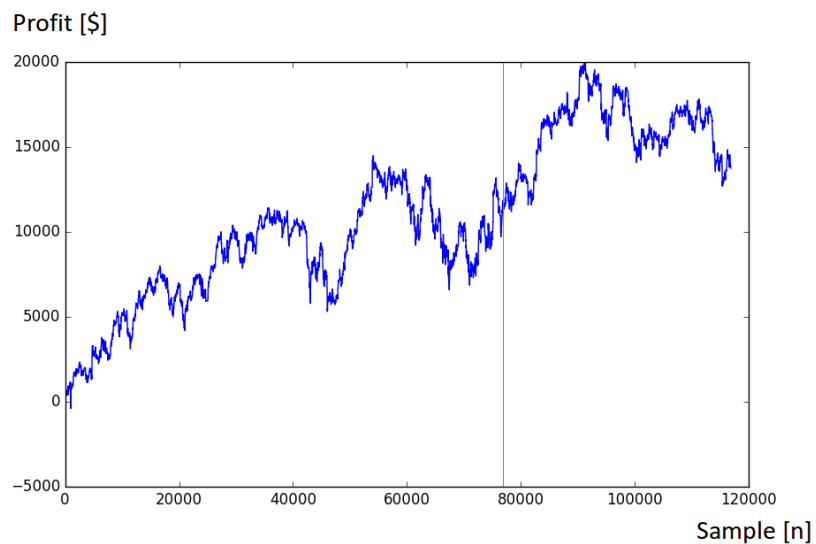


Obrázek 6.4: Regresní graf trénování pro EMD a TF (zleva).

Na obrázku 6.4 jsou dva regresní diagramy. Pravý (TF) je oproti levému znatelně roztahlější. To je způsobeno konstantou 5 (viz kapitola 5.3.1) při normalizaci výstupu pomocí SMA. Aktivační vrstva regresní sítě je na výstupu nastavena na lineární, takže by nemělo vadit, že hodnoty na výstupu se nacházejí v intervalu větším než $(-1, 1)$.



Obrázek 6.5: Obchodní výsledky jednoduché strategie na EMD.



Obrázek 6.6: Obchodní výsledky jednoduché strategie na TF.

Vertikální čára odděluje trénovací datovou část (nalevo) a testovací datovou část (napravo). Číslo na ose x představuje počet vzorků obchodních dat.

Metriky	Trénovací část	Testovací část
Hrubý zisk	4820\$	1966\$
Zisk po odečtení poplatků	-5185\$	-3444\$
Počet obchodů	2001	1082
Průměrná doba v pozici	13 min	13 min
Počet ziskových	848	412
Počet ztrátových	1153	670
Počet stop-lossů	689	443
Průměrný ziskový obchod	128.28\$	218.71\$
Průměrný ztrátový obchod	-90.17\$	-131.56\$
Sharpeho poměr	0.26	0.19
Max drawdown	4450\$	4100\$
Beta	0.23	0.19

Tabulka 6.4: Tabulka výsledků pro obchodní strategii na EMD.

Oba průběhy působí velmi nestabilním dojmem. Tomu odpovídá i maximální drawdown pro obě strategie. Jde vidět, že velké nárůsty a propady se navzájem střídají. Dalším problémem je, že obě obchodní strategie vykazují velký počet obchodů (transakcí). Uvedené výsledky jsou před odečtením poplatků. Poplatek je fixně 2.5\$ za kontrakt. Po odečtení poplatků je výdělek na testové části obchodních dat -5185\$ na EMD a -3444\$ na TF.

Metriky	Trénovací část	Testovací část
Hrubý zisk	10820\$	2940.0\$
Zisk po odečtení poplatků	-4450\$	-6050\$
Počet obchodů	3054	1798
Průměrná doba v pozici	13 min	14 min
Počet ziskových	1421	797
Počet ztrátových	1633	1001
Počet stop-lossů	1096	723
Průměrný ziskový obchod	162.6\$	219.1\$
Průměrný ztrátový obchod	-134.8\$	-171.5\$
Sharpeho poměr	0.25	0.22
Max drawdown	7925\$	6720\$
Beta	0.51	0.54

Tabulka 6.5: Tabulka výsledků pro obchodní strategii na TF.

Přestože regresní model na obou datových sadách vydělává a hodnota účtu má celkově stoupající tendenci, tak nevydělává dostatečně na to, aby to pokrylo i náklady za obchodování. Je nutné najít nějaké řešení, které sníží počet transakcí a zvýší výdělek. Tento efekt by mělo mít zvýšení přesnosti predikce, což by teoreticky mohlo jít přidáním dat na vstup neuronové sítě, která obsahuje další informace o trhu. Pokud jsou opomenuty informace, které může poskytnout fundamentální analýza, tak jde především o výstupy technické analýzy (technické indikátory).

6.2.2 Optimalizace použitých indikátorů

Za účelem nalezení vhodné kombinace indikátorů na vstupu byl proveden experiment, který měl najít tu nejlepší kombinaci. Myšlenka za tímto experimentem byla taková, že přidání všech indikátorů na vstup by mělo negativní dopad na kvalitu predikce, protože by vstup obsahoval redundantní části.

Vzhledem k tomu, že síť musela být v každé iteraci optimalizace znovu naučena s jinými vstupy, tak bylo vybráno pouze 8 indikátorů, aby optimalizace netrvala moc dlouho. Byly testovány všechny kombinace těchto osmi indikátorů, tedy včetně žádného až všech. Kombinací je celkem $2^8 = 256$.

Použité indikátory:

1. Volatilita za poslední 3 minuty
2. Volatilita za posledních 7 minut
3. EMA s intervalem 4 minuty
4. EMA s intervalem 7 minut
5. RSI s intervalem 12 minut
6. RSI s intervalem 27 minut
7. ROC s intervalem 7 minut
8. ROC s intervalem 12 minut

Iterativně byla procházena všechna čísla od 0 po 256. Binární reprezentace aktuálního čísla iterace představovala masku, která udávala jaké indikátory se mají použít při trénování sítě.

Sít byla následně záměrně testována na menší části dat z EMD, na které si předchodí síť bez indikátorů nevedla dobře. Jedním z důvodů bylo urychlení experimentu, protože síť nemusela být testována na celých datech. Druhým důvodem bylo, že se více ukáže vliv indikátorů na části, kde předchodí síť prodělávala.

Nejnižší dosažená hodnota byla -837\$, nejvyšší dosažená hodnota byla 1201\$ a průměrná hodnota byla 185.21\$. Hodnoty jsou uvedeny bez poplatků, protože počet transakcí byl ve všech případech podobný.

Případ	Vol 3	Vol 7	EMA 4	EMA 7	RSI 12	RSI 27	ROC 7	ROC 12
20 nejlepších	9	14	9	10	7	11	12	12
20 nejhorších	10	11	9	13	12	13	9	15

Tabulka 6.6: Počet výskytů jednotlivých indikátorů pro 20 nejlepších a 20 nejhorších výsledků.

Z tabulky nevyplývá, že by síť s použitím nějakého konkrétního indikátoru podávala lepší výsledky než ostatní. Největší rozdíl v počtu výskytů je mezi RSI 12 a ROC 7, ale může to být pouze náhoda. Pro ověření výsledku byl experiment zopakován se dvěma kombinacemi. První kombinace byla ta, která měla nejlepší výsledek a druhá, která měla nejhorší výsledek.

Nejlepšího výsledku dosáhla kombinace s klíčem 219, to odpovídá masce 11011011 (Vol 3, Vol 7, RSI 12, RSI 27, ROC 7, ROC 12). Nejhoršího výsledku dosáhla kombinace 144,

tedy maska 10010000 (Vol 3, EMA 7) Celý proces učení a testu byl pro tuto kombinaci zopakován 10x.

Kombinace	Průměr	Maximum	Minimum
Nejlepší (219)	341.25	958.0	-442.0
Nejhorší (144)	221.55	678.0	-619.0

Tabulka 6.7: Výsledky pro nejlepší a nejhorší kombinaci indikátorů po 10 opakováních.

Bohužel vliv indikátorů se dostatečně neprokázal, protože síť podává velmi rozdílné výsledky i pro stejné vstupní kombinace při trénování. Je možné, že za to mohla nevhodně vybraná testovací část obchodních dat (nedostatečná velikost). Taky se zdá, že více vstupů při trénování nemá na přesnost predikce nějaký negativní vliv. Naopak lepší výsledek podávala kombinace 219, která obsahuje indikátorů více, proto budou v příští části použity při trénování všechny indikátory.

6.2.3 Výsledky pro učení s indikátory

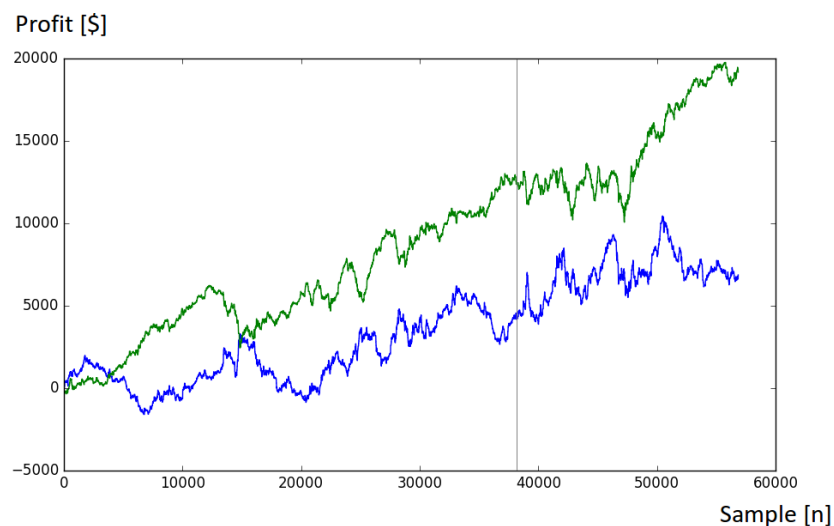
V této části je porovnána síť, která je učena s indikátory se sítí, která byla učena bez indikátorů. Dle závěrů z předchozí kapitoly jsou vybrány všechny indikátory.

Vstup	Prvků
Close	15
EMA 4	5
EMA 7	5
RSI 12	5
RSI 27	5
ROC 7	5
ROC 12	5
Volatilita 3	5
Volatilita 7	5
Celkem	47

Tabulka 6.8: Délka jednoho trénovacího vektoru.

Tabulka 6.10 představuje obsah jednoho vstupního vektoru regresní neuronové sítě. Jednoduchý model byl učen pouze na základě 15-ti historických CLOSE hodnotách.

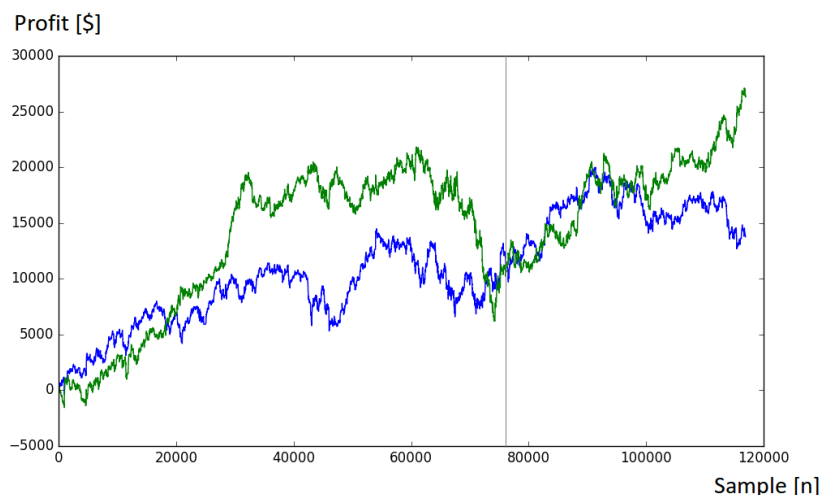
Jsou uvedeny pouze grafy výsledků a tabulky statistik. Regresní grafy a grafy učení byly podobné jako u jednoduché regresní neuronové sítě.



Obrázek 6.7: Výsledky strategie na EMD. Zelený průběh je model s indikátory.

Metriky	Trénovací část	Testovací část
Hrubý zisk	10710\$	6250\$
Zisk po odečtení poplatků	2945\$	1700\$
Počet obchodů	1553	910
Průměrná doba v pozici	14 min	15 min
Počet ziskových	778	447
Počet ztrátových	775	463
Počet stop-lossů	413	274
Průměrný ziskový obchod	94.47\$	134.18\$
Průměrný ztrátový obchod	-81.09\$	-116.04\$
Sharpeho poměr	0.78	0.72
Max drawdown	3800\$	3570\$
Beta	0.12	0.08

Tabulka 6.9: Tabulka výsledků pro obchodní strategii na EMD s indikátory.



Obrázek 6.8: Výsledky strategie na TF. Zelený průběh je model s indikátory.

Z grafů na obrázku 6.7 a 6.8 lze pozorovat, že výsledky pro modely, které při učení využívaly i indikátory, jsou na konci simulace lepší. Například obchodní strategie na EMD v testové části více než zdvojnásobila svůj výdělek. Na průběhu pro TF lze vidět na trénovací části sady velký nárůst stavu účtu oproti modelu, který nevyužíval indikátory. Bohužel pak došlo k razantnímu poklesu, ale na testové části datové sady si vedl obchodní model s indikátory lépe.

Metriky	Trénovací sada	Testovací sada
Hrubý zisk	10210\$	16100\$
Zisk po odečtení poplatků	-8690\$	1930\$
Počet obchodů	3780	2834
Průměrná doba v pozici	13 min	10 min
Počet ziskových	1718	1351
Počet ztrátových	2062	1483
Počet stop-lossů	1574	994
Průměrný ziskový obchod	163.1\$	124.0\$
Průměrný ztrátový obchod	-130.9\$	-102.1\$
Sharpeho poměr	0.478	0.53
Max drawdown	15620\$	5520\$
Beta	0.48	0.33

Tabulka 6.10: Tabulka výsledků pro obchodní strategii na TF s indikátory.

Celkový stav na testovací části obchodních dat pro model na EMD je 1700\$, to znamená, že model měl kladný výdělek i po odečtení poplatků za obchodování. Pro model TF je výdělek 11375\$, tento model byl ještě úspěšnější. Obchodoval téměř 2x delší dobu, ale přesto je výsledek více než 6-ti násobný. Přesto působí průběhy velmi nestabilním dojmem, což reflektuje i max. drawdown. Největší drawdown se nachází u modelu na TF v trénovací části datové sady. To, že se takový propad nachází v trénovací části datové sady může znamenat, že to, jak dobře si obchodní strategie vedly na testovacích částech dat. sady, může být pouze *štěstí*.

Testy na obou sadách byly při učení s indikátory lepší než referenční obchodní strategie.

Kapitola 7

Kombinace klasifikační a regresní sítě

Největším problémem předchozího modelu, který je založen na regresní neuronové síti je to, že provádějí příliš velký počet obchodních transakcí. To znamená, že výdělek sítě nedostačuje na pokrytí samotných poplatků za transakce.

Další nevýhodou je, že regresní model není dost dobře parametrizovatelný, takže není jednoduché nalézt způsob, kterým by šel regulovat počet provedených transakcí. Také se zdá, že predikce regresního modelu jsou zatíženy velkým šumem a regresní strategie často vstupují do nevhodných pozic, které nejsou dostatečně ziskové a dále zvyšují náklady na poplatky.

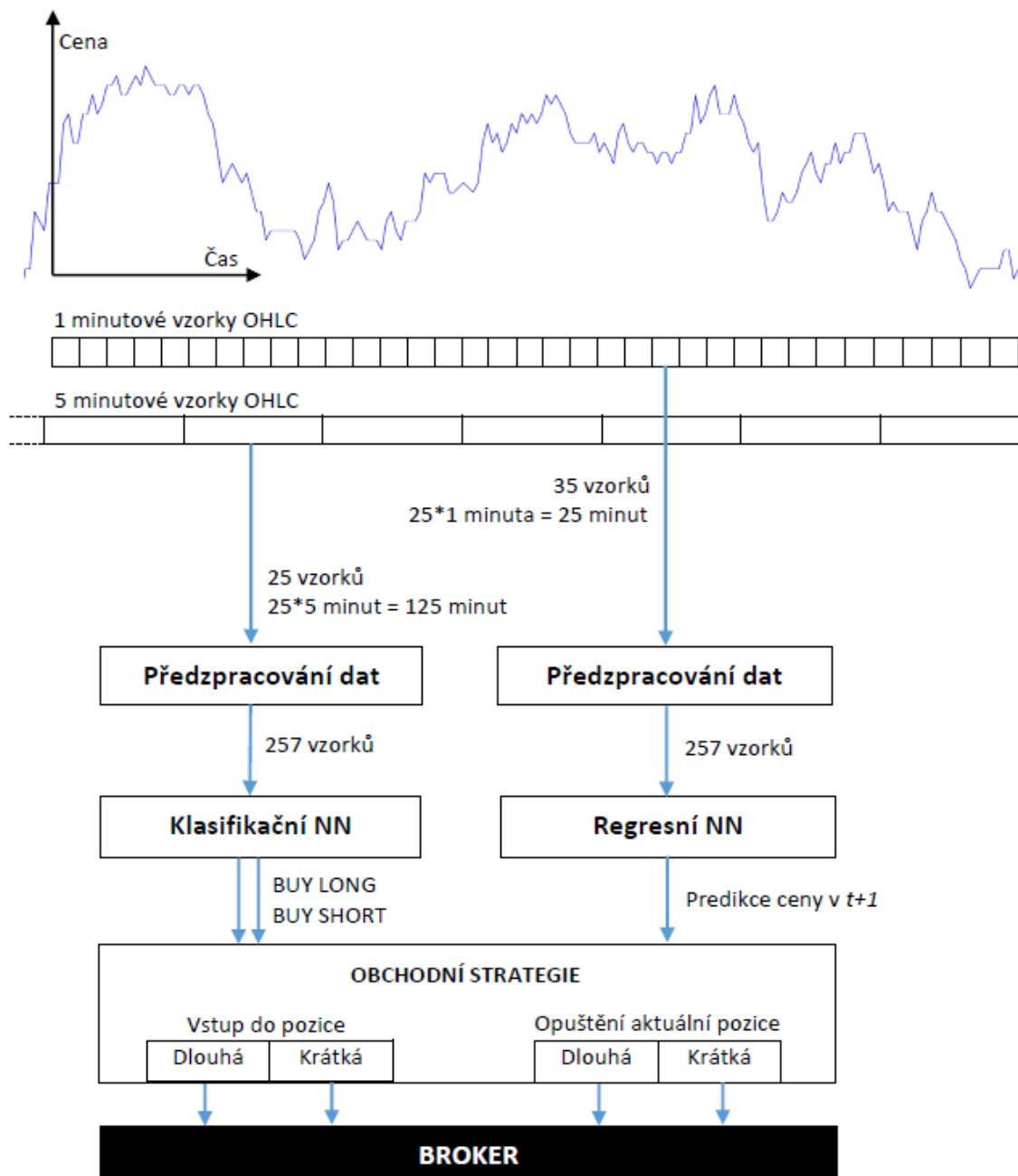
Z uvedených důvodů bude tato práce dále pracovat také s konceptem **klasifikačního modelu neuronové sítě**. Cílem tohoto modelu je přesnější určení vhodných okamžiků pro vstup do pozice, tak aby došlo k výraznému snížení provedených transakcí. Před trénováním tohoto modelu je potřeba vstupní vektory rozdělit do vhodných klasifikačních tříd. Bylo experimentováno s několika přístupy, jedním z nich, který byl nakonec použit, je následující: [7]

- Třída BUY LONG: Ano pokud je v následujících 5 vzorcích nárůst alespoň o 1,2 bodu (hodnota 1), jinak ne (hodnota -1)
- Třída BUY SHORT: Ano pokud je v následujících 5 vzorcích pokles alespoň o 1,2 bodu (hodnota 1), jinak ne (hodnota -1)

Další výhodou je, že počet vektorů zařazených do třídy BUY LONG a BUY SHORT lze ovlivnit parametrem, který určuje o kolik bodů má v následujících pěti vzorcích trh vzrůst (klesnout). Pokud bude požadovaný rozdíl moc malý, tak bude docházet opět k častému vstupování strategie do pozice. Pokud bude rozdíl moc velký, tak trénovacích vektorů zařazených do třídy BUY LONG (BUY SHORT) bude moc málo.

Klasifikační model pracuje s 5 minutovými daty

Důležitým rozdílem oproti regresnímu modelu je, že klasifikační model neuronové sítě pracuje nad daty s 5 minutovým rozlišením. OHLC 1 minutová data jsou podvzorkována na OHLC 5 minutová data. Zřejmým důvodem je, že model fungující na datech s větším intervalem bude vstupovat do menšího počtu pozic, než model pracující na minutových datech.



Obrázek 7.1: Schéma obchodní strategie, která se rozhoduje na základě klasifikační a regresní neuronové sítě.

Dalším důvodem je, že při podvzorkování dojde k omezení vlivu šumu v minutových datech. Naopak nevýhodou je, že model ztratí schopnost rychlé reakce a bude nutně reagovat pomaleji než model pracující na datech s menší periodou. Z toho důvodu bude obchodní strategie využívat také regresního modelu. S klasifikační sítí bylo experimentováno i na minutových datech, ale úspěšnost obchodního modelu nebyla tak velká jako v případě 5 minutových obchodních dat.

Klasifikační model neuronové sítě pracující na 5 minutových datech řídí vstup do pozice. V případě, že se obchodní strategie nachází v pozici (krátká nebo dlouhá), tak je tato pozice řízena regresním modelem neuronové sítě pracujícím na minutových datech. Tím by mělo dojít k požadovanému snížení počtu transakcí a optimálnímu určení okamžiku, kdy pozici opustit. Princip je ilustrován na obrázku 7.1.

7.0.1 Vstupy a topologie neuronové sítě

Využití indikátorů jako vstupu při trénování a predikci by mělo teoreticky zlepšit kvalitu výsledné predikce. Indikátory reprezentují některé důležité informace o trhu. Existuje spousta strategií, které využívají například RSI, EMA nebo Stochastic oscilátoru. Indikátory mohou reprezentovat závislosti trendů na volatilitě apod.

V předchozím experimentu se neprokázalo, že by nějaký indikátor nebo jejich kombinace dominovala nad nějakou jinou kombinací. Také se ale ukázalo, že více vstupů neovlivňuje schopnost predikce neuronové sítě negativně. Byly tedy přidány ještě další indikátory, které jsou rozepsány v tabulce 7.1. Timeframe jednoho vektoru je vždy 25, v 5-ti minutovém měřítku je to celkem 125 minut.

Vstup	Délka vektoru
Open	$15+15+15 = 45$
High	$15+15+15 = 45$
Low	$15+15+15 = 45$
Close	$15+15+15 = 45$
ROC(7, 14, 20)	$5+5+5 = 15$
RSI(7, 14, 20)	$5+5+5 = 15$
EMA(7, 14, 20)	$5+5+5 = 15$
Stochastik(17, 20, 26)	$10+10+10 = 30$
Volatilita(7, 20)	$1+1 = 2$
Celkem	257

Tabulka 7.1: Délka jednoho trénovacího vektoru.

Pro vstup OPEN je prvků celkem 45, protože počet použitých historických hodnot je 15 a jsou provedeny tři typy normalizace. Normalizace pomocí SMA, normalizace zlogaritmováním a transformace na uniformní rozdělení. Princip těchto transformací je popsán v kapitole 5.3.1. To samé platí i pro hodnoty HIGH, LOW a CLOSE. Hodnota v závorce u daného indikátoru v tabulce znamená období, za který je vyhodnocen. Pro každé vyhodnocené období daného indikátoru je na vstup sítě použito 5 jeho nejaktuálnějších hodnot.

Vrstva	Aktivační funkce	Neuronů	Dropout
Vstupní vrstva	lineární	16	0.23
1. Skrytá vrstva	tanh	13	0.13
2. Skrytá vrstva	tanh	8	0.10
Výstupní vrstva	tanh	2	0.00

Tabulka 7.2: Topologie klasifikační sítě.

S topologií sítě bylo experimentováno, ale nebyla nalezena taková topologie, která by fungovala lépe než topologie popsaná v tabulce 7.2.

Úpravy regresní sítě

Oproti regresní síti v kapitole 6.2 jsou přidány hodnoty OPEN, HIGH a LOW a také indikátory. Takže vstup je předzpracován stejným způsobem jako vstup pro klasifikační síť (tabulka 7.1). Timeframe pro jeden vstupní vektor je stejný jako předtím, a to 35 minut.

Výsledky trénování jsou téměř stejné jako v předchozí části, která se zabývala trénováním regresní sítě, proto zde nejsou uvedeny.

7.0.2 Úprava obchodní strategie

Výstupem klasifikační sítě jsou dvě hodnoty. První hodnota udává příslušnost do třídy BUY LONG, druhá ke třídě BUY SHORT. Hodnota +1 je maximální možná míra příslušnosti. V případě BUY LONG to značí silný signál k tomu, že v nejbližším vývoji trhu dojde ke zvýšení jeho hodnoty. Naopak velikost příslušnosti -1 znamená nulovou příslušnost daného vstupního vektoru ke třídě BUY LONG. To stejné platí i pro BUY SHORT. Výstupní hodnota sítě se nejspíše bude nacházet někde mezi těmito dvěma hodnotami. Jedním z možných parametrů, kterým bude možné ovlivnit výsledek testu na obchodních datech, bude volba prahu. Pokud bude výstupní hodnota sítě pro predikci míry příslušnosti do třídy BUY LONG větší než daný práh, bude to interpretováno tak, že právě klasifikovaná data do této třídy přísluší.

Princip obchodní strategie je stejný jako v případě samostatného regresního modelu. Je potřeba určit pravidla pro vstup do dlouhé a krátké pozice a také pravidla pro výstup z těchto pozic.

Pravidla pro vstup do dlouhé pozice využívají pouze predikce z klasifikační neuronové sítě, která pracuje na základě 5 minutových obchodních dat. Vzhledem k tomu, že klasifikační síť vrací míry příslušnosti pro třídy BUY LONG a BUY SHORT je vhodné využít obě tyto hodnoty.

Pravidlo pro vstup do dlouhé pozice:

$$y_{long} > 0.3 \wedge y_{short} < -0.5 \quad (7.1)$$

Pravidlo pro vstup do krátké police:

$$y_{short} > 0.3 \wedge y_{long} < -0.5 \quad (7.2)$$

Kde y_{long} je výstup klasifikační neuronové sítě značící míru příslušnosti ke třídě BUY LONG a y_{short} je míra příslušnosti ke třídě BUY SHORT. Pravidla pro opuštění pozice využívají i regresní neuronovou síť, která pracuje na základě minutových obchodních dat. Cílem je opustit pozici v ideálním okamžiku.

Pravidlo pro opuštění dlouhé pozice:

$$(y_{t+1} - y_t > 0.5 \wedge y_{long} < -0.3) \vee y_{short} > 0.0 \quad (7.3)$$

Pravidlo pro opuštění krátké pozice:

$$(y_{t+1} - y_t < -0.5 \wedge y_{short} < -0.3) \vee y_{long} > 0.0 \quad (7.4)$$

Kde y_{t+1} je predikovaná hodnota CLOSE v následujícím minutovém intervalu pomocí regresní neuronové sítě a y_t je aktuální hodnota CLOSE.

Dalším důležitým parametrem, který má velký vliv na to kolik je schopna strategie vydělat je stop-loss. Pravidlo pro stop-loss je možné nastavit fixně nebo dynamicky. V obchodním modelu s regresní sítí byl stop-loss nastaven fixně. Dynamický stop-loss je výhodný v tom, že bere ohled na aktuální situaci na trhu a nedochází k předčasnému opuštění pozice. Jednou z metod, která je použita v této obchodní strategii je nastavení stop-lossu pomocí Bollingerových pásem (sekce 3.1.2). Pokud se strategie nachází v dlouhé pozici, tak stop-loss je aktivován v případě, že aktuální hodnota trhu je menší než hodnota stop-loss nastavená při vstupu do pozice. U krátké pozice obdobně pokud je aktuální hodnota větší než nastavená hodnota stop-lossu.

Nastavení stop-lossu pro dlouhou pozici:

$$y_{stop-loss} = y_t - BB_{width} * 0.80 \quad (7.5)$$

Nastavení stop-lossu pro krátkou pozici:

$$y_{stop-loss} = y_t + BB_{width} * 0.80 \quad (7.6)$$

Kde $y_{stop-loss}$ je nastavená hodnota stop-lossu, y_t je aktuální hodnota trhu a BB_{width} je šířka Bollingerových pásem, tedy vzdálenost mezi nízkým a vysokým pásmem.

Posledním parametrem je nejmenší doba, po kterou musí strategie zůstat v pozici po vstupu do ní. Tento parametr má limitovat šum v regresní predikci a je nastaven na 5 minut. Čím je výsledek predikce z klasifikace přesnější, tím je možné tuto hodnotu zvyšovat, protože klasifikační síť predikuje nárůst v následujících 5-ti periodách. Vzhledem k tomu, že klasifikační síť pracuje na 5 minutových datech, tak to znamená predikci na 25 minut dopředu.

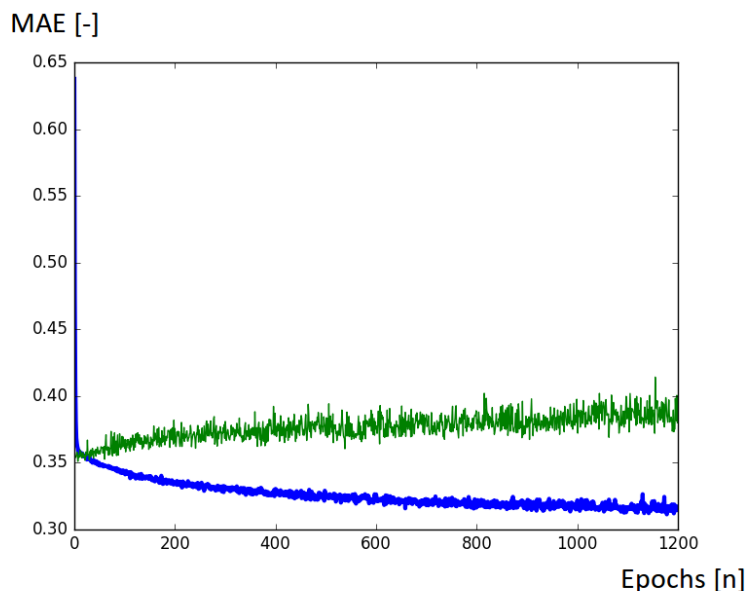
Prahové hodnoty a jiné parametry jako koeficient u stop-lossu jsou nastaveny intuitivně. Později budou optimalizovány pomocí evolučního algoritmu.

7.0.3 Učení klasifikační sítě

Tato část se zabývá učením klasifikační sítě. Ukázalo se, že učení je u této sítě daleko problematičtější než u regresního modelu. Často se stane, že se síť nic nenaučí. Respektive, že výstup pro všechny vstupní vektory konverguje k jedné hodnotě. Úspěšnost závisí výrazně i na topologii sítě, na nastavení dropout poměrů a regularizačních parametrech v jednotlivých vrstvách. Vypadá to, že úspěšnost trénování závisí také na počtu trénovacích vektorů. Topologie a parametry jsou popsány v tabulce 7.2. Tato část se zabývá učením pouze na datové sadě EMD, protože učení na TF je problematičtější a touto částí se zabývá podkapitola 7.0.4.

Když už se podařilo síť úspěšně natrénovat, tak se objevil další problém a to takový, že síť nepodává vždy stejně kvalitní výkony. Tento problém už se objevil při pokusu o optimalizaci

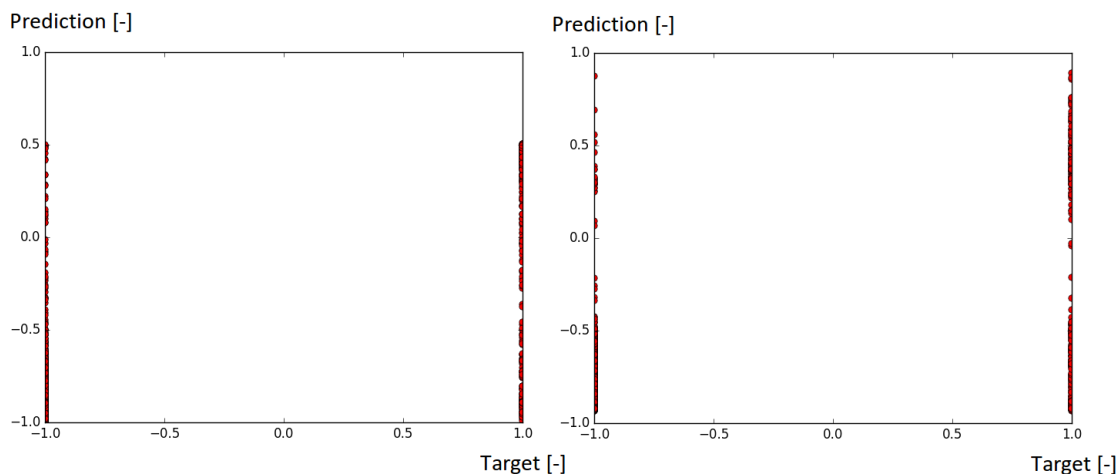
použitých indikátorů v čistě regresním modelu. V tomto případě je to ještě výraznější. Je to způsobeno tím, že inicializace¹ parametrů neuronové sítě před samotným trénováním není deterministická a vzhledem k tomu, že se jedná o velmi komplikovanou úlohu, tak je prostor řešení pravděpodobně velmi členitý. Kvalitu výsledného řešení nelze posoudit podle hodnoty objektivní funkce na trénovací ani validační sadě. V tomto ohledu jsou všechna řešení podobná. Ohodnocení kvality dané klasifikační neuronové sítě musí probíhat jako backtest na testovací části datové sady.



Obrázek 7.2: Průběh chyby při učení na trénovací části EMD. Zelená křivka je chyba na validační části trénovací datové sady.

Sít je trénována v 1200 epochách a trénovacích vektorů je 4844. Pokud byl počet epoch moc malý, tak se síť zpravidla nic nenaučila. I parametry dropout byly zvoleny experimentálně tak, aby síť fungovala. Pokud byl dropout moc malý, chyba na validační sadě vzrůstala rychleji než na obrázku 7.2 a síť na testovací sadě nepodávala dobré výsledky. Pokud byl dropout moc velký, síť se nebyla schopná nic naučit. Je možné, že pro jiné datové sady by bylo potřeba parametry vhodně upravit. Ideální by bylo použít nějaký evoluční nebo optimalizační algoritmus, který by hyperparametry vhodně určil.

¹Bylo experimentováno s různými typy inicializace, nakonec byla vybrána *he_normal* viz <http://keras.io/initializations/>



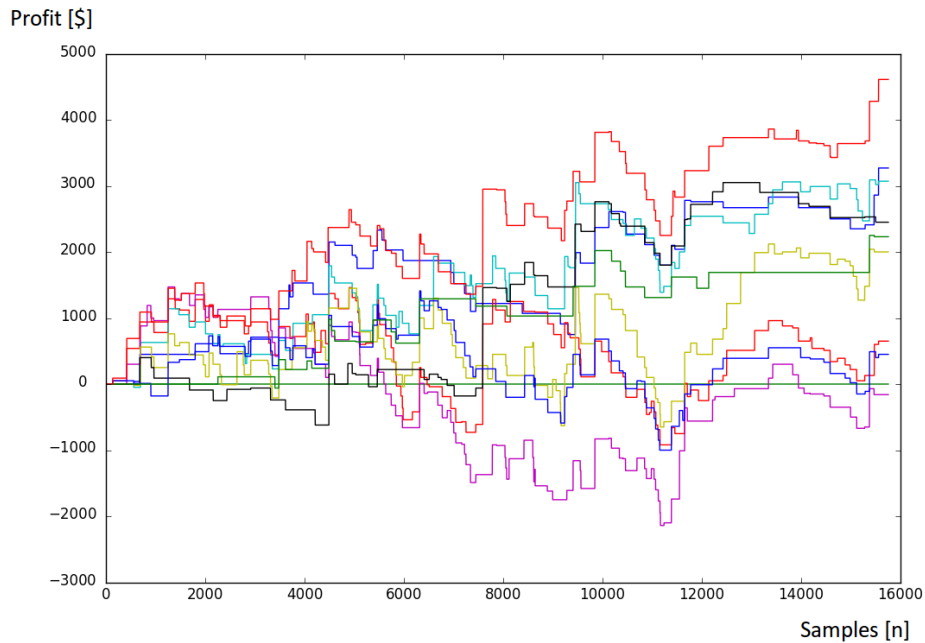
Obrázek 7.3: Regresní grafy pro klasifikační třídu BUY LONG a třídu BUY SHORT (zleva). BUY LONG nevyužívá celého oboru hodnot.

Regresní diagramy pro třídu BUY LONG a BUY SHORT (obrázek 7.3) ukazují, že výstup neuronové sítě pro třídu BUY LONG (oproti BUY SHORT) nevyužívá celého oboru hodnot výstupní aktivační funkce (tanh). Testování ukáže, zda tento zmenšený interval stačí.

Hledání nejlépe natrénovaného modelu

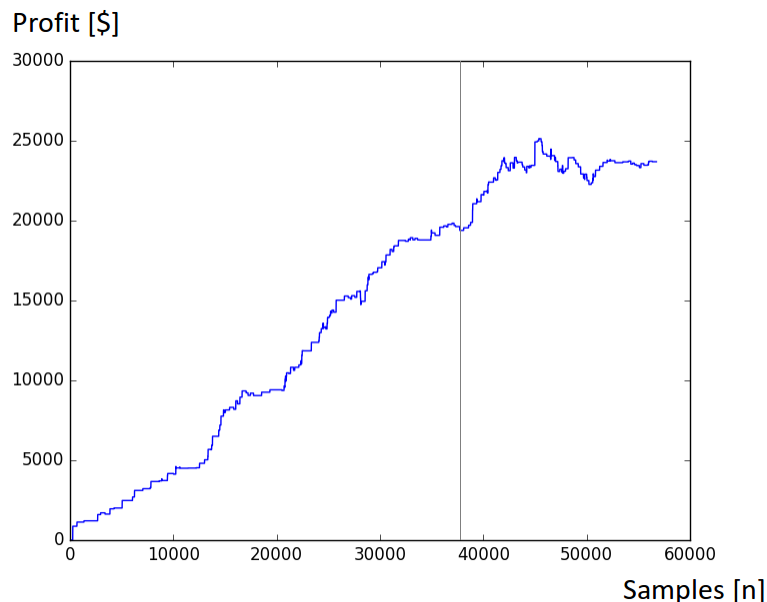
Protože výsledky trénování klasifikačního modelu nejsou vždy konzistentní, bylo učení 10x opakováno, aby mohl být následně vybrán model, který vykazoval nejlepší výsledky. Testování probíhalo na testové části datové sady a parametry simulovaného obchodování byly nastaveny tak, jak jsou popsány v kapitole 7.0.2.

Tato část nepopisuje učení regresní sítě, přestože je využita, tak jak je popsáno v obchodní strategii (kapitola 7.0.2).



Obrázek 7.4: Pokus o naučení klasifikační sítě na trénovací části dat EMD opakovaný 10x. Graf ukazuje výsledky pro zpětné testování na testové části datové sady.

Průměrně to vycházelo tak, že 25% ze všech trénovaných modelů se nic nenaučilo. Z grafu je vidět, že rozptyl ve výsledcích byl velký. Nejlepší síť byla profitabilní, naopak nejhorší mírně prodělávala. Jednotlivé průběhy jsou si tvarem docela podobné, ale horší sítě byly náchylnější k větším ztrátám a dílčí výtěžky nebyly tak vysoké jako u lepších modelů.



Obrázek 7.5: Průběh obchodování strategie využívající regresní a klasifikační sítě na EMD pro nejlepší model z předchozího pokusu o natrénování sítě.

Graf 7.5 představuje výsledek testování nejlepšího modelu z grafu 7.4. Testování obchodního modelu probíhalo na trénovací i testové části datové sady.

Metriky	Trénovací sada	Testovací sada
Hrubý zisk	19390\$	4300\$
Zisk po odečtení poplatků	18670\$	3725\$
Počet obchodů	144	115
Průměrná doba v pozici	65 min	47 min
Počet ziskových	97	52
Počet ztrátových	47	63
Počet stop-lossů	27	30
Průměrný ziskový obchod	260.0\$	268.3\$
Průměrný ztrátový obchod	-122.0\$	-153.2\$
Sharpeho poměr	1.31	0.40
Max drawdown	860\$	2870\$
Beta	0.44	0.40

Tabulka 7.3: Tabulka výsledků pro obchodní strategii na EMD

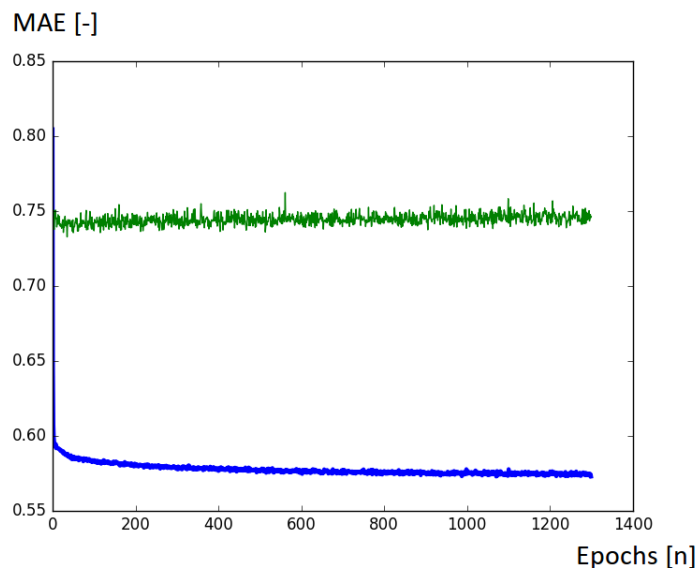
Z tabulky 7.3 je zřejmé, že počet transakcí se podařilo oproti čistě regresnímu modelu výrazně snížit. A to tak, že na testovací datové sadě skončila strategie se ziskem +3725\$. Jde vidět, že na testovací datové sadě si síť nevede tak dobře jako na trénovací části, ale to se dá očekávat. Vykazuje větší drawdown a také horší Sharpeho poměr a větší část obchodů byla ztrátová oproti ziskovým. Naštěstí ziskový obchod měl větší průměrný výdělek, než ztrátový obchod. Zdá se, že čím jsou data dále od předělu mezi testovací a trénovací částí, tím jsou výsledky labilnější.

Byl vybrán **nejlepší model**, protože v kapitole 8 bude tento model použit k obchodování na živých datech. Tento model byl vybrán na základě testování na testové části datové sady, další data nejsou k dispozici. Na základě těchto možností není jiný způsob jak vybrat lepší model.

7.0.4 Učení klasifikační sítě na TF

U této datové sady se nepodařilo najít vhodné hyperparametry neuronové sítě jako v předchozím případě. Bylo experimentováno s různými nastaveními, ale přesto nebylo učení sítě tak úspěšné jako u datové sady EMD. Následující experimenty nakonec využívají stejné topologie jako model v předchozí kapitole, viz tabulka 7.2. Tato část nepopisuje učení regresní sítě, přestože je využita, tak jak je popsáno v obchodní strategii (kapitola 7.0.2).

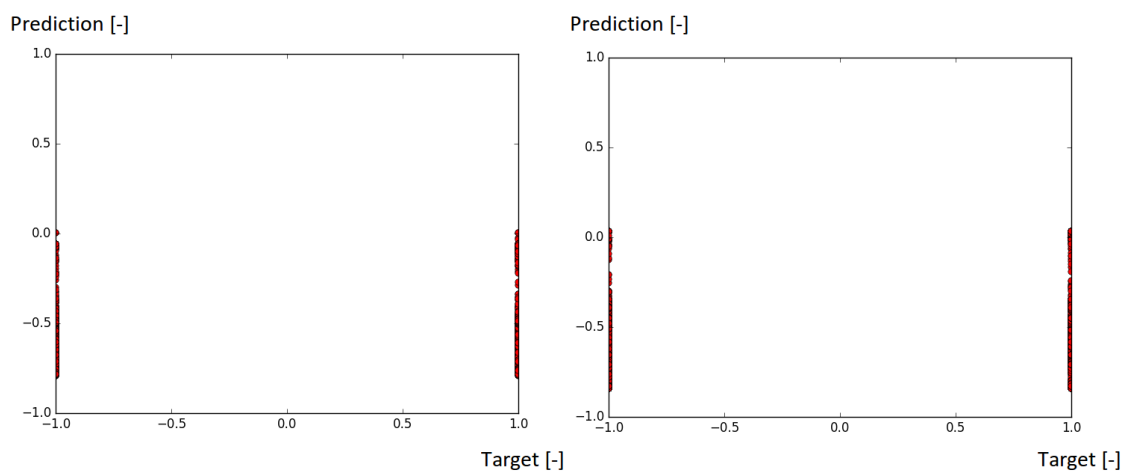
Následující experimenty popisují výsledky učení při klasickém rozdělení datové sady na trénovací (2/3) a testovací část (1/3) a výsledky při učení s rozdělením 1/2 a 1/2, protože bylo zjištěno, že v tomto případě, když je síť učena na méně datech, jsou výsledky lepší. V obou případech byl průběh chyby při trénování velmi podobný, viz průběh pro trénování na klasickém rozdělení v grafu 7.6.



Obrázek 7.6: Průběh chyby při učení na trénovací části TF (2/3 dat). Zelená křivka je chyba na validační části trénovací sady a modrá křivka je chyba na trénovací části.

Výsledky pro učení s typickým rozdělením trénovací a datové sady

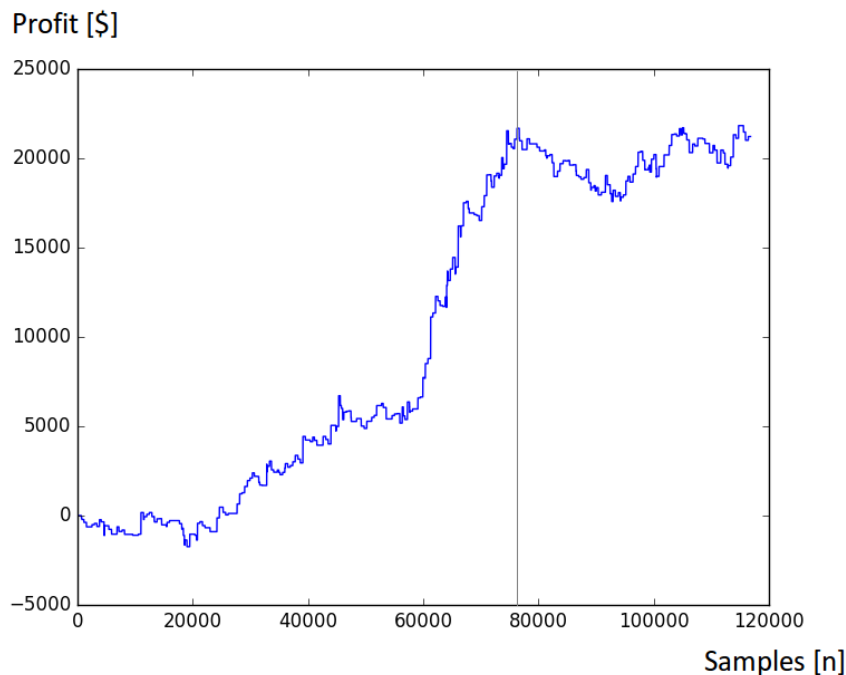
V tomto případě nastal problém takový, že výstupní hodnoty klasifikační sítě nevyužívají celý obor hodnot výstupní aktivační funkce $(-1,1)$, což lze jasně vidět na regresních grafech pro obě klasifikační třídy (BUY LONG a BUY SHORT), viz grafy na obrázku 7.7.



Obrázek 7.7: Regresní grafy pro klasifikační třídu BUY LONG a třídu BUY SHORT (zleva). Narozdíl od sítě učené na EMD nevyužívá tato síť celý rozsah výstupních hodnot, jak by měla (viz graf 7.3 pro porovnání).

Epoch učení bylo 1300 a trénovacích vektorů bylo 10600. Bylo experimentováno s daleko menším i daleko větším počtem trénovacích epoch, ale to na učení nemělo pozitivní vliv. Z regresních grafů je jasné, že nastavení parametrů obchodní strategie z části 7.0.2 nebude

v tomto případě fungovat.



Obrázek 7.8: Průběh obchodování strategie na TF při normálním rozdělení datové sady na trénovací a testovací část.

Parametry pravidel (především prahové hodnoty) obchodní strategie byly voleny tak, aby pokud možno reflektovaly rozsahy výstupních hodnot této klasifikační neuronové sítě. Upraveny byly tak, že od výchozích parametrů týkajících se klasifikační sítě (specifikovaných v kapitole 7.0.2) byla odečtena hodnota 0.5. Výsledky ukáží, zda síť úspěšně modeluje chování trhu.

Metriky	Trénovací sada	Testovací sada
Hrubý zisk	23060\$	230\$
Zisk po odečtení poplatků	21970\$	-360\$
Počet obchodů	218	118
Průměrná doba v pozici	128 min	117 min
Počet ziskových	108	48
Počet ztrátových	110	70
Počet stop-lossů	65	39
Průměrný ziskový obchod	488.0\$	386.3\$
Průměrný ztrátový obchod	-269.0\$	-249.2\$
Sharpeho poměr	0.61	0.13
Max drawdown	3650\$	6080\$
Beta	0.18	0.15

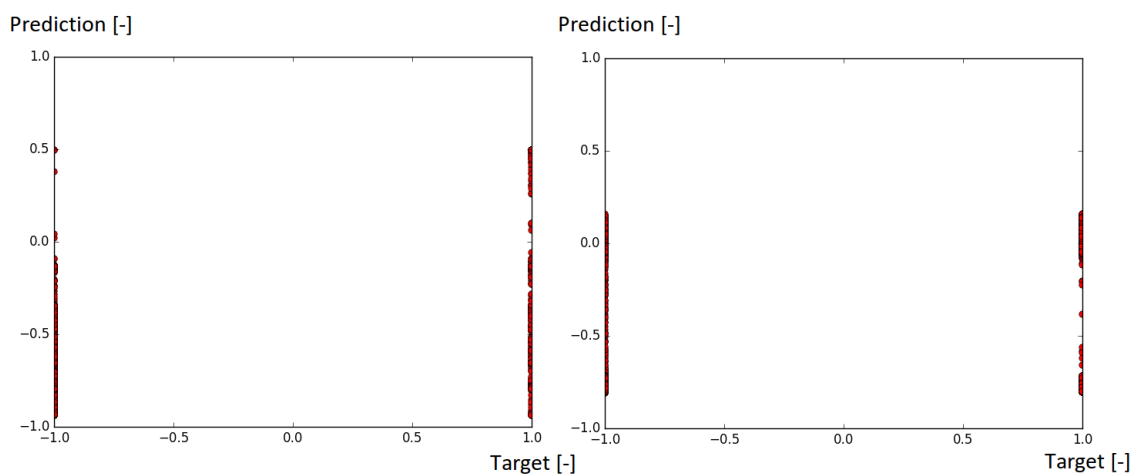
Tabulka 7.4: Tabulka výsledků pro obchodní strategii na TF při normálním rozdělení trénovací a testovací části.

Závěr je takový, že tento model si (i přes ne úplně optimální výsledky trénování) vede

docela obstojně. Sice ne tak dobře, aby byl zisk na testovací části kladný (konkrétně -360\$), ztráta není vysoká. Kontinuální propady střídají nárazové nárůsty. Parametry obchodní strategie byly voleny intuitivně, výsledek by byl pravděpodobně výrazně lepší, pokud by byly algoritmicky optimalizovány.

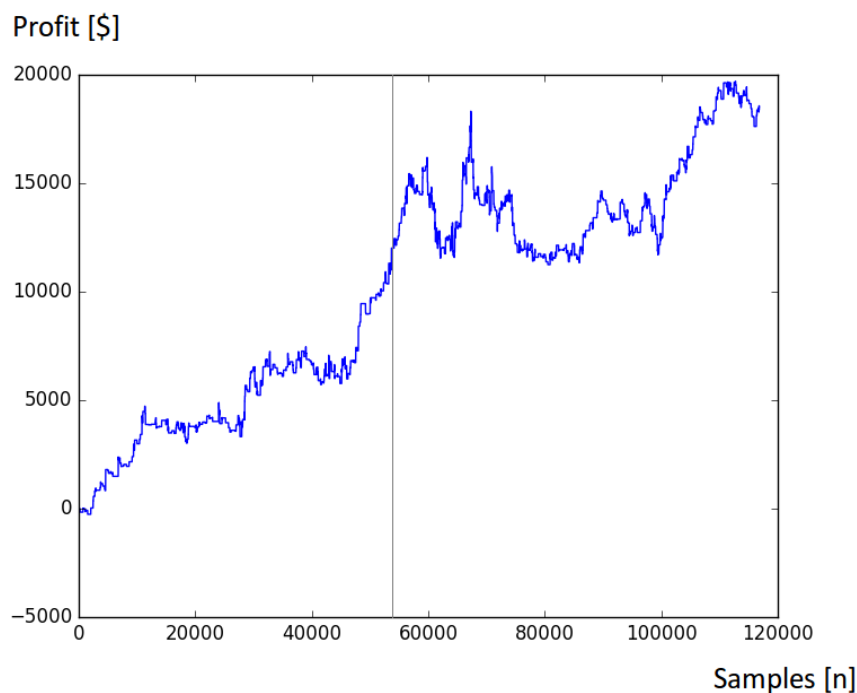
Výsledky pro učení s menší trénovací částí

V tomto experimentu byla zmenšena trénovací část datové sady TF, protože se ukázalo, že to má na učení pozitivní vliv. Rozdělení je takové, že trénovací část tvoří první polovina dat (místo 2/3) z datové sady TF a zbytek dat tvoří testovací část. Tato menší trénovací část má 7472 trénovacích vektorů (oproti předchozím 10600 vektorům). Z toho, že jsou výsledky trénování lepší (graf 7.9), je možné usoudit, že nastavení topologie a parametrů učení sítě není pro větší počet trénovacích vektorů vhodné.



Obrázek 7.9: Regresní grafy pro klasifikační třídu BUY LONG a třídu BUY SHORT (zleva).

Parametry obchodní strategie byly také upraveny pro vhodný rozsah této sítě podobně jako v předchozím případě, ale od výchozích parametrů byla odečtena pouze hodnota 0.2.



Obrázek 7.10: Průběh obchodování obchodní strategie na TF se zkrácenou trénovací částí.

Metriky	Trénovací sada	Testovací sada
Hrubý zisk	11550\$	6680\$
Zisk po odečtení poplatků	7560\$	4760\$
Počet obchodů	798	384
Průměrná doba v pozici	33 min	35 min
Počet ziskových	378	185
Počet ztrátových	420	199
Počet stop-lossů	110	53
Průměrný ziskový obchod	209.9\$	184.6\$
Průměrný ztrátový obchod	-161.4\$	-137.1\$
Sharpeho poměr	0.30	0.40
Max drawdown	1820\$	6340\$
Beta	0.11	0.25

Tabulka 7.5: Tabulka výsledků pro obchodní strategii na TF se zkrácenou trénovací částí.

Čistý zisk na testovací části je 4760\$, což je mnohem lepší zisk než v předchozím experimentu s větší trénovací datovou sadou. Ze Sharpeho poměrů a drawdownu vyplývá, že strategie se chová docela labilně, což je zřejmé i z pohledu na průběh obchodování.

Zhodnocení dosažených výsledků na TF

Z výsledků vyplývá, že klasifikační síť byla schopna úspěšně se něco z trénovací sady naučit. Síť s menším počtem trénovacích vektorů (7472 oproti 10600) vykazuje lepší výsledky. Lze proto usoudit, že volba topologie a nastavení sítě není vhodné pro větší počet trénovacích

vektorů. Je pravděpodobné, že vhodnějším nastavením hyperparametrů sítě (topologie a trénovací parametry) by mohlo být dosaženo kvalitnějších výsledků učení (regresní grafy) i výsledků na testové části datové sady. Problém je, že parametrů je mnoho a není jasné, který parametr má jaký vliv. Parametry by mohly být vybírány náhodně nebo intuitivně, nejvhodnější by bylo použít nějaký optimalizační algoritmus. Ten by vyžadoval spoustu výpočetního výkonu a času, protože trénování této sítě trvalo při dané konfiguraci téměř hodinu a testování několik desítek minut.

7.1 Optimalizace parametrů obchodní strategie

Obchodní strategie je řízena pravidly, které jsou založeny na výstupech regresní a klasifikační sítě. Tyto pravidla obsahují určité prahové hodnoty, které rozhodují, jak bude pravidlo vyhodnoceno. Původní prahové hodnoty v kapitole 7.0.2 byly určeny intuitivně. Jde o následující prahové hodnoty (resp. parametry):

Vstup do pozice:

$$y_{pos} > p_{enter_pos} \wedge y_{neg} < p_{enter_neg} \quad (7.7)$$

Opuštění pozice:

$$(y_{t+1} - y_t > p_{exit_diff} \wedge y_{pos} < p_{exit_pos}) \vee y_{neg} > p_{exit_neg} \quad (7.8)$$

Nastavení stop-lossu:

$$y_{stop-loss} = y_t \pm BB_{width} * p_{stop_coef} \quad (7.9)$$

Pravidla není třeba dělit na dlouhá a krátká, protože jsou symetrická. V případě pravidla pro dlouhou pozici y_{pos} (pozitivní) znamená predikci pro třídu BUY LONG a y_{neg} (negativní) znamená predikci pro třídu BUY SHORT. Naopak v případě pravidla pro krátkou pozici je y_{neg} (negativní) predikce pro třídu BUY LONG a y_{pos} (pozitivní) je predikce pro třídu BUY SHORT.

Pak symboly p označují příslušné parametry, které mohou být zapsány jako vektor:

$$(p_{enter_pos}, p_{enter_neg}, p_{exit_diff}, p_{exit_pos}, p_{exit_neg}, p_{stop_coef}, p_{min_pos_time})$$

Parametr $p_{min_pos_time}$ řídí minimální počet minut, který musí strategie strávit v pozici před opuštěním.

7.1.1 Optimalizace pomocí PSO na EMD

Optimalizace bude probíhat pomocí evolučního algoritmu, konkrétně byl zvolen PSO (Particle Swarm Optimization) implementovaný v knihovně DEAP². PSO je optimalizační technika inspirovaná pohybem hejna při vyhledávání potravy. Parametry algoritmu jsou uvedeny v tabulce 7.6. Fitness funkcí je jednoduše dosažený výtěžek v obchodní simulaci, od kterého jsou odečteny poplatky za obchodování, aby byly penalizovány řešení, které vstupují do pozic příliš často. Tato fitness funkce bude maximalizována.

²DEAP <https://deap.readthedocs.io/en/master/>

Parametr	Hodnota
Velikost populace	5 částic
Počet generací	50
Dimenzionalita částice	7
Maximální pozice	+1
Minimální pozice	-1

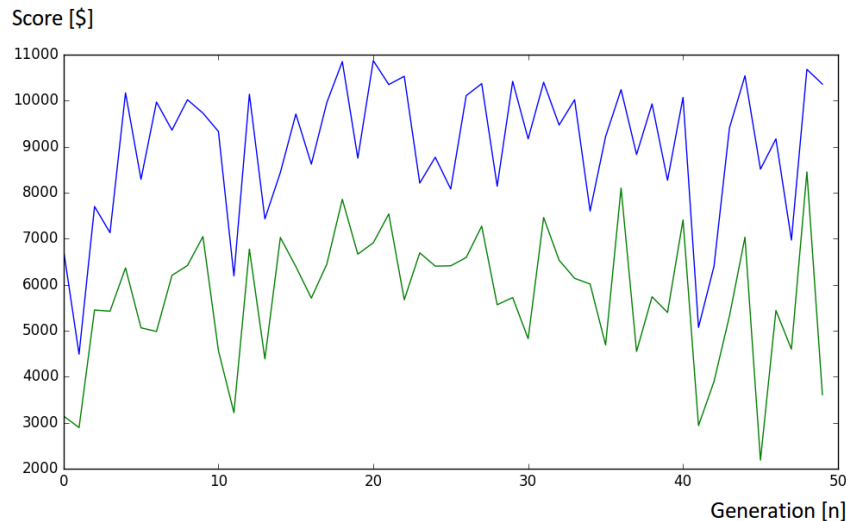
Tabulka 7.6: Nastavení optimalizace pomocí PSO.

Pro každý parametr bylo určeno zobrazení z intervalu $(-1,1)$ (min, max poloha dle tabulky) na vhodný interval, který by pro parametr co nejlépe reprezentoval možný rozsah použitelných hodnot a tím i urychlil proces optimalizace.

Parametr	Interval
p_{enter_pos}	$\langle -1, 1 \rangle \rightarrow \langle -0.5, 0.5 \rangle$
p_{enter_neg}	$\langle -1, 1 \rangle \rightarrow \langle -0.5, 0.5 \rangle$
p_{exit_diff}	$\langle -1, 1 \rangle \rightarrow \langle 0.5, 2.0 \rangle$
p_{exit_pos}	$\langle -1, 1 \rangle \rightarrow \langle -0.5, 0.5 \rangle$
p_{exit_neg}	$\langle -1, 1 \rangle \rightarrow \langle -0.5, 0.5 \rangle$
p_{stop_coef}	$\langle -1, 1 \rangle \rightarrow \langle 0.5, 2.5 \rangle$
$p_{min_pos_time}$	$\langle -1, 1 \rangle \rightarrow \langle 0, 20 \rangle$

Tabulka 7.7: Transformace intervalu pro daný optimalizovaný parametr.

Optimalizace probíhala na testovací části datové sady symbolu EMD. Neprobíhala i na TF, protože na této datové sadě byly problémy s učním, viz podkapitola 7.0.4. Použitý model pro optimalizaci byl vybrán na základě několika učení (viz graf 7.4). Graf průběhu pro maximální a průměrnou hodnotu populace je na obrázku 7.11.



Obrázek 7.11: Průměrná fitness populace (zelená) a maximální fitness z populace (modrá).

Z průběhu optimalizace lze pozorovat, že v 50 generacích průměrná hodnota fitness populace nekonvergovala, ale to není takový problém, protože optimalizace našla několik

úspěšných řešení. Z tohoto pohledu je zajímavá maximální dosažená fitness v rámci dané populace. Z grafu lze odhadnout, že teoretický strop byl nejspíše někde nad hodnotou 10000\$. Minimální dosažená hodnota byla -9790\$ a maximální dosažená byla 10870\$.

Analýza nalezených řešení

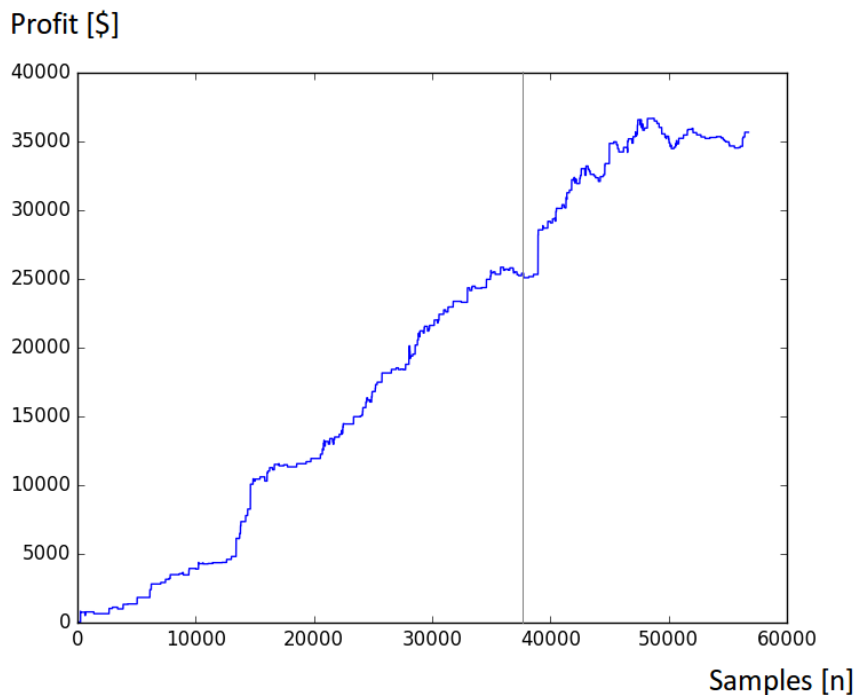
Ze všech nalezených řešení bylo vybráno 5 nejlepších, ty dosáhly hodnot 10870\$, 10850\$, 10680\$, 10540\$ a 1053\$. Parametry ve všech 5 nejlepších řešeních vykazují podobné rysy, což je dobré znamení. Tabulka 7.8 obsahuje průměrnou, maximální a minimální hodnotu pro každý jednotlivý parametr. Tyto hodnoty jsou spočítány pro 5 nejlepších řešení.

Parametr	Průměrná hodnota	Největší	Nejmenší
p_{enter_pos}	-0.05	0.03	-0.14
p_{enter_neg}	0.17	0.18	0.14
p_{exit_diff}	0.78	1.16	0.58
p_{exit_pos}	0.20	0.42	0.00
p_{exit_neg}	0.40	0.48	0.30
p_{stop_coef}	1.17	1.21	1.14
$p_{min_pos_time}$	6	7	6

Tabulka 7.8: Analýza pro 5 nejlepších řešení nalezených PSO.

Jednotlivé parametry nevykazují velký rozptyl. Nejmenší rozptyl vychází u $p_{min_pos_time}$ a p_{stop_coef} . Největší rozdíly jsou u p_{exit_diff} a p_{exit_pos} . Nejlepší řešení s hodnocením 10870 může být vybráno jako nastavení pro další experimenty. Parametry nejlepšího řešení jsou velmi blízké průměrným hodnotám uvedeným v tabulce 7.8.

7.1.2 Výsledky pro optimalizovaný obchodní model na EMD



Obrázek 7.12: Průběh obchodování pro optimalizovaný obchodní algoritmus na EMD

Metriky	Trénovací sada	Testovací sada
Hrubý zisk	25090\$	10580\$
Zisk po odečtení poplatků	24275\$	9985\$
Počet obchodů	163	119
Průměrná doba v pozici	79 min	70 min
Počet ziskových	111	62
Počet ztrátových	52	57
Počet stop-lossů	28	26
Průměrný ziskový obchod	296.6\$	344.35\$
Průměrný ztrátový obchod	-150.7\$	-189.0\$
Sharpeho poměr	1.71	0.98
Max drawdown	920\$	2200\$
Beta	0.10	0.14

Tabulka 7.9: Tabulka výsledků pro optimalizovanou obchodní strategii na EMD.

Tento model s optimalizovanými parametry obchodní strategie dosáhl více než dvakrát lepšího výsledků, než model s parametry před optimalizací (výsledky v tabulce 7.3). Přestože jde vidět, že ke konci simulace na testové části jsou výsledky mírně labilní, byl model na této části celkově v zisku 9985\$. Drawdown je na testové části dvojnásobný oproti trénovací části. Ale hodnoty Sharpeho poměru jsou velmi dobré (0.98 na testové části).

Kapitola 8

Živé obchodování

Jelikož obchodní model popsaný v kapitole 7 (viz schéma 7.1), který využívá kombinace klasifikační a regresní sítě podává celkem dobré výsledky, bude vyzkoušen na živém obchodování. Brokerů, kteří nabízejí API, aby bylo možné jednoduše obchodovat algoritmicky, není kupodivu mnoho. Jedním z mála známějších brokerů je Interactive Brokers (dále jen IB). Tento broker nabízí celkem standardní funkci a to možnost vyzkoušet si živé obchodování na demo účtu, takže není potřeba žádného finančního vkladu.

Mezi nevýhody demo účtu patří následující omezení:

- Obchodní data jsou simulovaná, trh by se měl chovat podobně jako reálný, ale data nejsou stejná.
- Uživatelský účet není perzistentní, při každém novém sezení je uveden do výchozího nastavení.
- Chyby a nestabilní chování, které se v plné verzi nevyskytuje. Příkazy nemusí být vždy vykonány apod.
- Menší uživatelská podpora.

Asi největší problém je nemožnost získat historická data, přestože Interactive Brokers API tuto možnost nabízí, tak je možné získat data pouze za dva dny z minulosti. Navíc tyto obchodní data absolutně neodpovídají tickovým aktuálním hodnotám. Z těchto důvodů není možné model naučit na simulovaných a aktuálních datech, proto bude použit model naučený na datech EMD, který je popsán v předchozích kapitolách. Z toho vyplývá, že obchodování bude probíhat na symbolu EMD. Obchodování na TF není na IB ani možné, protože nebylo možné na těchto datech provést příkaz na vstup do pozice (pravděpodobně jedna z chyb demo účtu).

IB API

Zasílání příkazů a příjem dat probíhá přes IB API. IB nabízí API pouze v C++ a Javě, ale existují i neoficiální knihovny pro Python. Zde je využita knihovna swigibpy. Knihovna swigibpy je vytvořena pomocí nástroje Swig. Tento nástroj umožňuje jednoduše "zabalit" C++ funkce tak, aby byly použitelné i v Pythonu. Knihovna swigibpy¹ je docela nízkoúrovňová,

¹Swigibpy <https://github.com/Komnomnomnom/swigibpy>

takže jsou využity funkce², jež jsou postaveny na swigibpy, se kterými se pracuje jednodušeji.

Teprve nad těmito funkcemi je postaveno vlastní zjednodušené API, které poskytuje rychlý přístup k datům a jednoduché zasílání příkazů.

IB API komunikuje s IB serverem buď skrz negrafický IB Gateway nebo přes grafický program Trader Workstation (oba programy jsou dostupné na oficiální stránce IB). Výhodou IB Gateway je, že je méně náročný na systémové zdroje a také je údajně stabilnější. Naopak výhodou Trader Workstation je právě uživatelské rozhraní, které umožňuje sledovat aktuální stav trhu, vykreslovat grafy, sledovat stav jednotlivých pozic, zadávat příkazy a tak dále.

Implementace

Zásadním rozdílem oproti předchozí práci je zpracování dat. Data zde zpracovávaná jsou ticková. API umožňuje definovat, s jakou periodou budou data přijímána. Tato perioda je nastavena na 5s, to znamená, že každých 5s jsou přijmuta ticková data. Protože regresní síť pracuje na minutových datech a klasifikační na datech s rozlišením 5 minut, tak jsou ticková data navzorkována s periodou jedné minuty. Obchodní model data obdrží až v okamžiku, kdy jich je nashromážděno dostatečné množství dle zvoleného časové rámce (timeframe).

- Regresní síť pracuje na minutových datech. Časový rámec je 35 minut.
- Klasifikační síť pracuje na 5-ti minutových datech. Časový rámec je 25. V minutách je to tedy $25 \cdot 5 = 125$ minut.

Když je minutových dat dostatek, jsou podvzorkována na 5-ti minutová data. Obchodní model tedy obdrží data vždy až 125 minut po spuštění programu.

Algoritmus obchodní strategie pak funguje standardně, na základě výsledků z modelů neuronových sítí vyhodnotí pravidla, na jejich základě se rozhodne případně vstoupit nebo vystoupit z pozice.

Jak už bylo zmíněno výše, jedním z omezení demo účtu je, že stavu účtu není perzistentní. Z toho vyplývá, že je třeba tento stav ukládat nějak externě. Stav účtu a příslušné příkazy jsou ukládány do externího CSV souboru. Zpracovaná data jsou také ukládána do CSV souboru, aby je bylo možné v následující kapitole spolu s výsledky vykreslit.

8.1 Výsledky

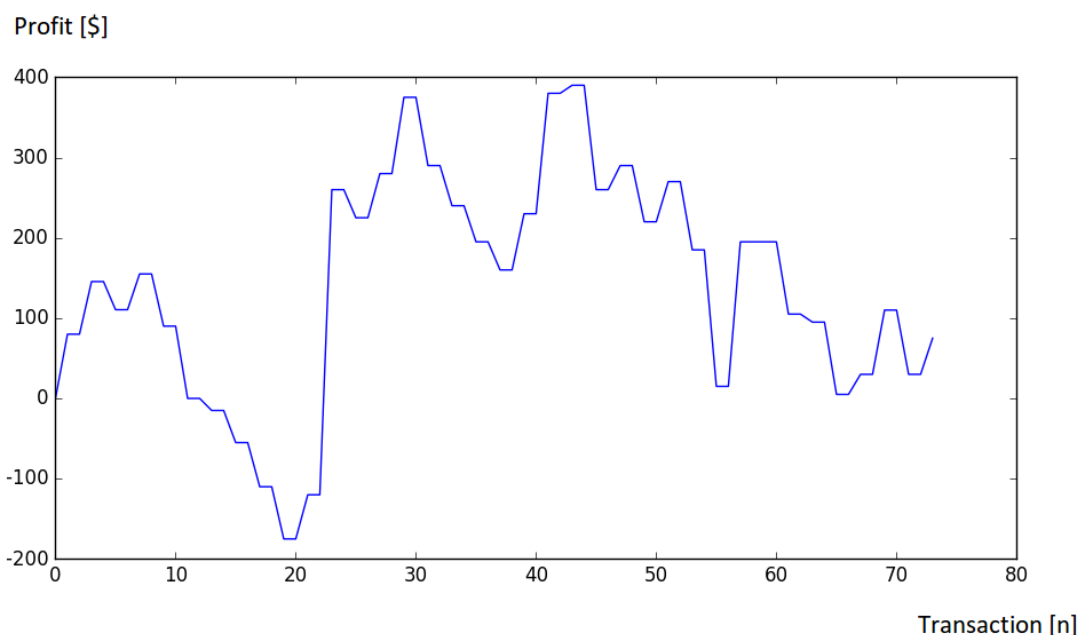
Následující výsledky jsou za období od 9.5.2016 do 18.5.2016. Zaznamenaná obchodní data jsou v grafu 8.1 a průběh obchodování je v grafu 8.2. Dlouhá diagonální čára v obchodních datech představuje víkend, kdy nebyl trh sledován. Nasbíraných minutových vzorků bylo přes 5000.

²<http://qoppac.blogspot.cz/2014/03/using-swigibpy-so-that-python-will-play.html>



Obrázek 8.1: Vývoj EMD v době kdy na něm byla testována obchodní strategie.

Celkový výsledek obchodního modelu při obchodování na demo účtu je záporný. Po odečtení poplatků za obchodování je zisk -100\$, protože transakcí proběhlo 72 (36 obchodů). Je vidět, že průběh obchodování nemá žádný trend. Maximální dosažená hodnota stavu účtu byla zhruba 400\$ a minimální -175\$.



Obrázek 8.2: Průběh zisku při testování obchodní strategie na demo účtu. Symbol EMD.

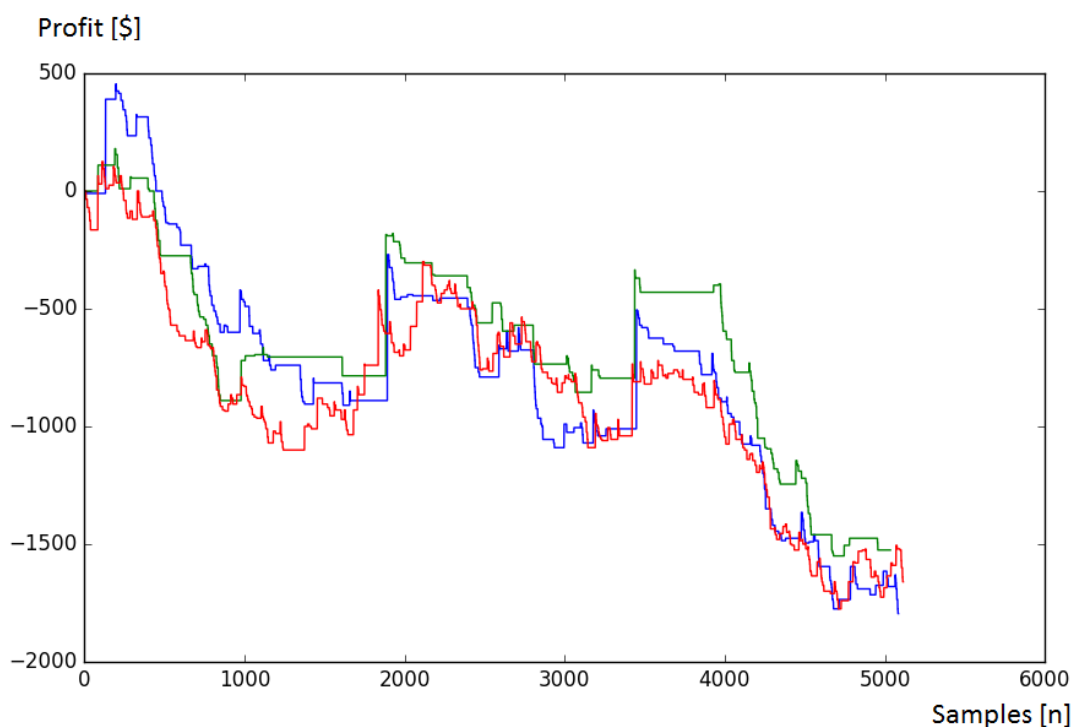
Obchodní strategie si v živém obchodování nevedla moc dobře, ale to se dalo očekávat, protože trh je na demo účtu IB nějakým nespécifikovaným způsobem simulován. A hlavně data, na kterých byl obchodní model trénován (EMD) jsou téměř 3 roky stará, protože aktuální nebyla k dispozici.

Tento experiment byl zajímavý hlavně jako "proof of concept" a nebyl zamýšlen jako

měřítko úspěšnosti obchodního modelu. Myšleno tak, že bylo ukázáno, že s dostatečným kapitálem není dnes problém nasadit na trh automatizovanou obchodní strategii implementovanou pomocí moderních knihoven v populárním programovacím jazyku usnadňujícím rychlost vývoje takových aplikací.

Výsledky pro referenční obchodní strategii

Pro srovnání je na obrázku 8.3 přiloženo zpětné testování na nasbíraných datech (graf 8.1) pro referenční obchodní strategii (viz část 6.1) s několika zvolenými obdobími (20, 50 a 100 minut). Graf obsahuje výtěžky bez započtených poplatků za transakce. Graf má na ose y použity jiné hodnoty než graf 8.2 (vzorky, na kterých bylo obchodováno místo jednotlivých transakcí), ale vzhledově jsou si v tomto ohledu grafy ekvivalentní.



Obrázek 8.3: Průběh zisku při testování referenční strategie na demo účtu. Symbol EMD. Zelená je SMA pro období 100, modrá SMA 50 a červená SMA 20.

Jde vidět, že ani jedna ze strategií nebyla úspěšná. V tomto ohledu byla použitá strategie založená na regresní a klasifikační síti mnohem lepší. Nejlíp dopadla referenční strategie s SMA pro období 100 minut. Ta dosáhla ztráty -1525\$ a -\$815 za obchodování, celkem tedy -2340\$. Obchodů bylo provedeno 163, z toho 20 ziskových a 143 ztrátových.

Kapitola 9

Závěr

V první části byly diskutovány některé základní pojmy, které se týkají obchodování na burze a algoritmického obchodování jako je technická analýza, technické indikátory a obchodovatelné trhy. Čtvrtá kapitola se zabývala teoretickými základy ohledně neuronových sítí.

V praktické části byly na začátku specifikovány dvě datové sady minutových obchodních dat, na kterých jsou trénovány a testovány představené obchodní strategie. Obě datové sady jsou minutová obchodní data z rozdílných E-Mini Futures trhů.

Obchodní strategie je definována především svými pravidly, na základě kterých se rozhoduje, zda vstoupit do krátké či dlouhé pozice. Pokud se v pozici nachází, tak se na základě pravidel rozhoduje, kdy z ní vystoupit. U každé obchodní strategie jsou tato pravidla definována.

Praktická část této práce se dá dále rozdělit na tři části. V první je specifikována referenční obchodní strategie založená na SMA, dále je představena jednodušší strategie, která pracuje pouze s jednou regresní neuronovou sítí. Obchodní model se rozhoduje pouze na základě predikce z tohoto modelu. Jelikož je tento model díky poplatkům za obchodování v konečném důsledku ztrátový, tak je snaha o zlepšení vlastností tohoto modelu tím, že je vstup regresní sítě rozšířen o technické indikátory. Tento model je vyhodnocen v kapitole 6.2.3. Model s indikátory podává lepší výsledky než referenční obchodní strategie.

Testování obchodního modelu probíhá jako simulace obchodování na obchodních datech, vlastnosti tohoto testování jsou popsány v kapitole 7.0.2. Výstupem z testování je průběh zisků a tabulka, která obsahuje konečný zisk po odečtení poplatků za obchodování, různé metriky a statistiky.

Protože výsledky obchodního modelu založeného na regresi nejsou uspokojivé (často vstupuje do obchodů a průběh obchodování je labilní), je v druhé části představen model kombinující klasifikační neuronovou síť a regresní síť z předchozího modelu. Klasifikační síť pracuje na 5-ti minutových datech. Vstup do pozice (dlouhé nebo krátké) řídí tato klasifikační síť a v okamžiku, kdy se obchodní strategie nachází v pozici, tak je výstup z této pozice řízen současně regresní i klasifikační sítí. Regresní síť se v této strategii nachází proto, aby mohla strategie reagovat na změny rychleji, jelikož pracuje na minutových datech. Pravidla obchodní strategie jsou o něco složitější než v předchozím případě a obsahují určité parametry, které jsou v kapitole 7.1 optimalizovány.

Poslední experiment se zabývá nasazením obchodního modelu kombinujícího klasifikační a regresní síť na živé obchodování simulovaného trhu pomocí demo účtu u Interactive Brokers. Výsledek obchodování není ziskový, ale to se dalo očekávat, protože data, na kterých byl model trénován, nebyla aktuální a navíc demo účet pracuje pouze se simulovanými

daty. Přesto byl vytvořený obchodní model úspěšnější než referenční obchodní model, který skončil ve větší ztrátě.

Přestože jsou výsledky testování na pokročilém modelu s EMD daty velmi dobré, trénování na datech TF ukázalo, že natrénovat takový model nemusí být jednoduché a záleží na počtu trénovacích vektorů a dalších parametrech dané datové sady. Je velmi pravděpodobné, že při vhodnější volbě topologie sítě a trénovacích parametrů by bylo trénování úspěšnější. Jenže neuronová síť obsahuje obrovské množství nastavitelných parametrů, u kterých není jasný vliv na konečné schopnosti sítě. První věc v dalším vývoji by měl být robustní algoritmus využívající evoluční algoritmy, který by umožnil tyto parametry jako topologie a parametry trénování vhodně optimalizovat. Jeden z problémů je, že by tento optimalizační algoritmus potřeboval velký výpočetní výkon a spoustu času, protože trénování jednoho modelu trvá na systému s CPU Intel i7 a GPU Nvidia GTX 860 více než hodinu.

Zajímavé by bylo také prozkoumat vliv dalšího trénování (obnovy) sítě s pomocí aktuálních dat v průběhu testování nebo živého obchodování. Proběhly určité experimenty, které nejsou v této práci popsány, ale vyplývalo z nich, že pokud je dat příliš málo, tak jsou další obchodní výsledky mnohem horší, než kdyby k novému trénování nedošlo. Ale je zřejmé, že při dostatečně dlouhém obchodování na živých datech by se tato situace musela řešit, protože modely neuronových sítí by časem zastarávaly a tím by se stávaly méně úspěšnými.

Dalším možným směrem vývoje je návrh modelu, který by byl schopný pracovat s tickovými daty. Možností aplikací neuronových sítí existuje v této oblasti velká řada. Bylo by zajímavé také prozkoumat schopnosti rekurentních neuronových sítí. Výhodné je, že v algoritmické obchodní strategii se tyto modely neuronových sítí dají libovolně kombinovat. Nutností je pak kvalitní testování navrženého modelu.

Literatura

- [1] Antoni Wysocki, M. L.: An Investment Strategy for the Stock Exchange Using Neural Networks. *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*.
- [2] Aronson, D.: *Evidence-Based Technical Analysis: Applying the Scientific Method and Statistical Inference to Trading Signals*. Wiley, 2006.
- [3] Dvořák, R.: *Trading strategie*. CPRESS, 2008, ISBN 78-80-251-2240-2.
- [4] Elder, A.: *Sell and Sell Short*. Impossible, 2014.
- [5] Forex-Zone: *Forex*. Grada, 2011.
- [6] Iebling Kaastra, M. B.: Designing a neural network for forecasting financial and economic time series. *Neurocomputing 10 Elsevier*.
- [7] Kordos, M.; Cwiok, A.: *Intelligent Data Engineering and Automated Learning - IDEAL 2011: 12th International Conference, Norwich, UK, September 7-9, 2011. Proceedings*, kapitola A New Approach to Neural Network Based Stock Trading Strategy. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ISBN 978-3-642-23878-9, s. 429–436.
- [8] Nison, S.: *Cesta k úspěchu na burzovních trzích*. Impossible, 2013.
- [9] Orr, G. B.; Mueller, K.-R. (editoři): *Neural Networks : Tricks of the Trade*. Springer, 1998.
- [10] Pavlát, V.: *Globální finanční trhy*. EUPRESS, 2013, ISBN 978-80-7408-076-0.
- [11] Rojas, R.: *Neural Networks - A Systematic Introduction*. Springer-Verlag, 1996.
- [12] Sharpe, M.: Lognormal model for stock prices.
- [13] Tan, C.: An Artificial Neural Networks Primer with Financial Applications Examples in Financial Distress Predictions and Foreign Exchange Hybrid Trading System. *School of Information Technology, Bond University*.
- [14] Turek, L.: *První kroky na burze*. CPRESS, 2008, ISBN 978-80-251-1915-0.