



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DETEKCE ELLIOTOVÝCH VLN

ELLIOT WAVE DETECTION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MAREK KALETA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PATRIK PETŘÍK

BRNO 2011

Abstrakt

Práce se zabývá detekcí Elliottových vln, statistického nástroje k popisu a předvídání trhu. Práce obsahuje návrhy možných metod jejich detekce a jejich zhodnocení. Z možných metod je pak rozpracována detekce impulsů pomocí umělých neuronových sítí, konkrétně množiny vícevrstvých perceptronů zapojených do jednoduché committee machine. Výstupem práce je program detekující Elliottovy vlny. Program hledá vlny přímo na vstupní časové řadě pomocí neuronových sítí a pak na tvoří hierarchickou strukturu vln.

Abstract

This work deals with Elliott wave detection, which are statistical tool used to describe financial makret cycles and predict market trends. The work proposes methods to detect Elliott Waves and evaluetes them. From several methods of Elliott wave detection, Committee machines of multilayer perceptrons are used. Result of this work is a program, which detect Elliott impulses on input signal and builds hierarchy of Elliott waves.

Klíčová slova

Elliottovy vlny, umělé neuronové sítě, percetron, vícevrstvý perceptron, detekce vzorů, klasifikace

Keywords

Elliott wave, Artificial Neural Network, Perceptron, Multilayer Perceptron, pattern recognition, classification

Citace

Marek Kaleta: Detekce Elliotových vln, bakalářská práce, Brno, FIT VUT v Brně, 2011

Detekce Elliotových vln

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Patrika Petříka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Marek Kaleta
18. května 2011

Poděkování

Děkuji Ing. Patriku Petříkovi za metodickou, pedagogickou a odbornou pomoc i další rady při zpracování mé bakalářské práce.

© Marek Kaleta, 2011.

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 Elliottovy vlny | 4 |
| 2.1 Základní pojmy | 4 |
| 2.2 Pravidla Elliottových vln | 5 |
| 2.2.1 Monovlny | 5 |
| 2.2.2 Polyvlny | 6 |
| 2.2.3 Impuls | 6 |
| 2.2.4 Korekce | 7 |
| 2.3 Fibonacciho řada, zlatý řez a Elliottovy vlny | 11 |
| 2.3.1 Poměry výšek vln v impulsu | 11 |
| 2.3.2 Poměry výšek vln v korekci | 11 |
| 2.4 Stupně vln | 11 |
| 2.5 Současné systémy pro detekci Elliottových vln | 11 |
| 3 Neuronové sítě a jejich použití při klasifikaci | 12 |
| 3.1 Neuronové sítě | 12 |
| 3.1.1 Neuronová síť | 12 |
| 3.1.2 Comittee machines | 14 |
| 3.2 Užití neuronových sítí pro klasifikaci | 14 |
| 4 Možnosti detekce Elliottových vln | 15 |
| 4.1 Detekce na základě hierarchického systému pravidel | 15 |
| 4.1.1 Pravidla pro identifikaci jednotlivých segmentů Elliottových figur | 15 |
| 4.1.2 Provedení vyhledávání | 17 |
| 4.1.3 Hodnocení metody | 17 |
| 4.2 Detekce charakteristických částí figur | 17 |
| 4.2.1 Vytváření figur z charakteristických částí pomocí rozhodovacího stromu | 18 |
| 4.2.2 Vytváření figur z charakteristických částí pomocí comittee machine | 18 |
| 4.3 Detekce větších celků a jejich postupné dělení | 18 |
| 4.4 Závěr | 19 |
| 5 Detekční systém | 20 |
| 5.1 Návrh systému | 20 |
| 5.1.1 Trénovací modul a generátor Elliottových vln | 20 |
| 5.1.2 Multilayer Perceptron | 21 |
| 5.1.3 Vstup a preprocessing | 21 |
| 5.1.4 Comittee machine | 22 |

| | | |
|----------|---|-----------|
| 5.1.5 | Tvorba hierarchie | 22 |
| 5.1.6 | Výstupní modul | 24 |
| 5.1.7 | Uživatelské rozhraní | 24 |
| 5.2 | Možnosti ovlivnění funkce systému | 26 |
| 5.2.1 | Perceptrony | 26 |
| 5.2.2 | Parametry filtrování impulsů v <i>committee machine</i> | 26 |
| 5.2.3 | Parametry tvorby hierarchie | 26 |
| 6 | Výsledky | 27 |
| 6.1 | Detekce testovacích impulsů pomocí různých sad sítí | 27 |
| 6.2 | Detekce impulsů na časových řadách finančního trhu | 27 |
| 7 | Závěr | 31 |
| A | Manuál | 33 |
| A.1 | Instalace a požadavky na funkci programu | 33 |
| A.2 | Ovládání programu v příkazové řádce | 33 |
| A.3 | Režim textových příkazů | 34 |
| A.4 | Typické použití programu | 34 |
| A.5 | Požadavky na vstupní data | 34 |
| A.6 | Snímek obrazovky programu | 34 |

Kapitola 1

Úvod

Tato práce se zabývá detekcí Elliottových vln pomocí umělých neuronových sítí. Cílem práce je vytvořit program, který na zadané časové řadě správně detekuje Elliottovy vlny. Elliottovy vlny jsou nástrojem pro analýzu chování finančních trhů. Výsledky analýzy lze použít pro popis minulého chování trhu nebo pro předpověď pravděpodobného chování trhu v budoucnosti.

V práci jsou navrženy různé postupy, které mohou vést k detekci a klasifikaci Elliottových vln, jsou uvedeny jejich výhody a nevýhody a následně zvolen postup detekce impulsů. Je vytvořen program, který pomocí tohoto postupu detekuje impulsy v zadané řadě a s jejich pomocí určuje další vlny.

První část práce je zaměřena na úvod do teorie Elliottových vln s přihlédnutím k jejich detekci a klasifikaci, dále na úvod do problematiky neuronových sítí. V další části je navržen postup detekce vln a popsán program, který detekci provádí. V závěru jsou zhodnoceny výstupy programu a diskutován přínos a další možný vývoj práce.

Kapitola 2

Elliottovy vlny

Elliottovy vlny jsou českým názvem pro jeden z možných analyticko-statistických přístupů k popisu trendů na burzovních trzích. Tato metoda se zakládá na pracích Ralpha Nelsona Elliotta.

Systém Elliottových vln dnes staví na hypotéze fraktálního trhu a používá některé části teorie chaosu, ale byl vytvořen čistě na základě statistiky Ralphem Nelsonem Elliottem během 30. a 40. let dvacátého století. [6] Při porovnání grafů vývoje cen upozoroval určité zákonitosti, které sledují veškeré pohyby na trhu. Svou práci rozvinul a později vydal jako [2]. Teorii Elliottových vln dále rozšířil zejména v 80. letech 20. století Robert Prechter spolu s A. J. Frostem jako [3].

Základem teorie je pozorování, že jednotlivé finanční pohyby na trhu nejsou zcela náhodné, ale mají určitou vnitřní strukturu, která se navíc opakuje a to jak postupně v čase, tak při přibližování grafu. Lze říci, že vlny mají fraktální strukturu [6]

Tato kapitola popisuje Elliottovy vlny podle [6] a [3], terminologie je přebrána z [6], je omezena na vlastnosti vln z hlediska jejich detekce a klasifikace, jejich vlastnosti z hlediska předpovědi dalšího chování trhu po ukončení vlny, případně další tvarování probíhající vlny nejsou uvedeny.

2.1 Základní pojmy

Vlna je základní jednotka Elliottovy teorie. Vlnou rozumíme pohyb ceny v čase v určitém směru. Pohyb může být směrový (rostoucí nebo klesající) a nesměrový. Podle rozsahu (stupně) rozlišujeme monovlny, polyvlny, multivlny a makrovlny, podle postavení ve vlně vyššího stupně hybné vlny a korekční vlny.

Výška vlny je absolutní hodnota rozdílu ceny v počátečním a koncovém bodě vlny.

Délka vlny je doba jejího trvání, tj. čas od začátku do konce vlny.

Stupeň vlny je vágní pojem vyjadřující velikost vlny. Vlna s větším pohybem a v delším čase je vlnou vyššího stupně.

Monovlna Je nejmenší úsek grafu, ze kterého se skládají veškeré větší útvary popsané teorií Elliottových vln. Tento úsek bývá monotonní.

Polyvlna se skládá z několika monovln.

Multivlna se skládá z několika polyvln nebo multivln nižšího stupně, na svém stupni je chápána jako jedna vlna.

Makrovlna se skládá z několika multivln nebo makrovln nižšího stupně, na svém stupni je chápána jako jedna vlna.

Figura se skládá z několika vln, jde tedy o obecné označení polyvlny, multivlny a makrovlny. Je složena z vln nižšího stupně a splňuje určitá pravidla.

Hybné vlny se pohybují ve stejném směru jako vlna o stupeň větší.

Korekční vlny se pohybují v opačném směru než vlna o stupeň vyšší.

Strukturní štítek slouží k zařazení cenového pohybu do příslušné třídy. Zapisuje se dvojtečkou a počtem vln ve formaci. Používá se zejména :3 a :5.

Postupový štítek slouží k značení každé vlny uvnitř figury. Části hybné figury se značí číslovkami, části korekční figury písmeny.

2.2 Pravidla Elliottových vln

2.2.1 Monovlny

Podle [6] je základem monovlny je takový souvislý úsek grafu, který ve všech bodech svírá s osou x úhel alespoň 45° a je v celé délce rostoucí, nebo klesající. Ostatní úseky lze nazvat neutrální. Zda budou neutrální úseky považovány za samostatné vlny nebo přiřazeny k jedné ze sousedících monovln určuje pravidlo neutrality.

Pravidlo proporce

Každá Elliottova figura závisí na poměru mezi cenou a časem. Analyzovaný graf [6] doporučuje transformovat tak, aby nejvýznamnější směrový pohyb v grafu svíral s osou x úhel 45° .

Pravidlo neutrality

[6] rozděluje zbylé úseky takto: Pro neutrální úsek, jež se nachází mezi monovlnami opačných směrů se používá aspekt 1 pravidla neutrality. Podle něj se v případě, že neutrální úsek leží mezi rostoucí a klesající vlnou, přiřadí se tento k rostoucí vlně vyjma případu, kdy by monovlna tvořená neutrálním úsekem spolu s klesající monovlnou byla nižší než 61,8% vlny rostoucí. Pak se neutrální úsek přiřadí ke klesající vlně.

V případě, kdy neutrální úsek leží mezi klesající a rostoucí vlnou, přiřadí se ke klesající vyjma případu, kdy by monovlna tvořená neutrálním úsekem a rostoucí monovlnou byla nižší než 61,8% vlny klesající. Pak se neutrální úsek přiřadí k rostoucí vlně.

Pro neutrální úsek ležící mezi monovlnami stejného směru se používá aspekt 2. Neutrální úsek je považován za samostatnou vlnu, pokud trvá déle než předchozí nebo následující monovlna.

Rozsah monovlny

Úsek grafu je považován za monovlnu, pokud

- tento klesající úsek prorazí minimum předchozího rostoucího úseku
- tento rostoucí úsek prorazí maximum předchozího klesajícího úseku
- rostoucí úsek následující za tímto klesajícím prorazí maximum předchozího rostoucího úseku
- klesající úsek následující za tímto rostoucím prorazí minimum předchozího klesajícího úseku.

2.2.2 Polyvlny

Polyvlny se skládají z několika monovln, tvoří *figury*. Hybnou figurou je *impuls*, základní korekční figurou je *trojúhelník* a *rovina*. Každá figura se skládá ze segmentů, které mohou být jednotlivými monovlnami, nebo se z monovln skládat.

Pravidlo podobnosti a rovnováhy

Figuru tvoří vlny stejného stupně. Vlny stejného stupně jsou si podobné v čase nebo ceně. Dvě vlny jsou si podobné v čase, pokud delší z nich není více než trojnásobně delší od druhé vlny, obdobně vlny podobné v čase se neliší výškou více než trojnásobně.

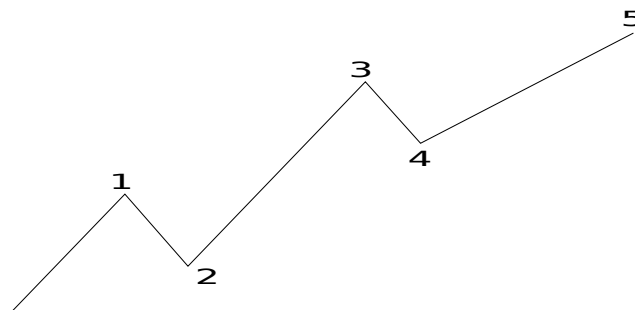
Pravidlo času

Žádné tři po sobě jdoucí vlny stejného stupně nejsou stejně dlouhé. Nejčastěji jsou dlouhé v poměru Fibonacciho čísel. Pokud jsou dvě za sebou jdoucí vlny stejně dlouhé, třetí je mnohem kratší nebo mnohem delší. Pokud je druhá vlna mnohem delší než první, pak je třetí 61,8% nebo 161,8% první.

2.2.3 Impuls

Impuls může být považován za základní vlnu Elliottovy teorie. Skládá se z 5. vln nižšího stupně, jsou značeny 1 až 5. Idealizovaný průběh impulsu zobrazuje obrázek 2.1.

Obrázek 2.1: Idealizovaný průběh hybného impulsu



Aby skupina vln mohla být považována za impuls, musí splňovat následující podmínky.

1. Skládá se z 5 segmentů, 1., 3. a 5. mají stejný směr a 2. a 4. k nim opačný. Segmenty jsou tvořeny vlnami stejného stupně.
2. Druhá vlna je nižší než první.
3. Třetí vlna je vyšší než druhá.
4. Čtvrtá vlna je nižší než třetí, pokud nejde o závěrečný impuls.
5. Pátá vlna je delší než čtvrtá.
6. Pátá vlna je vyšší než 38,2% čtvrté.
7. Třetí vlna není nejnižší z 1., 3. a 5.

Pokud pátá segment není vyšší než čtvrtá, jde o selhání 5. vlny.

První, třetí, nebo pátá vlna impulsu může být prodloužena. Taková vlna je vysoká alespoň 161,8% druhé nejvyšší vlny.

Prodloužení

Prodloužení vlny znamená, že je tato podstatně delší nebo složitější, než ostatní vlny ve stejné figuře. Nejčastější prodlouženou vlnou bývá u impulsu 3. vlna.

Podrozdělení

Podrozdělení vlny znamená, že tato vlna se dělí do více monovln než ostatní vlny, neboli vlna se skládá z vln více stupňů než ostatní vlny. Podrozdělení se často vyskytuje u prodloužené vlny. Příklad impulsu s prodlouženou podrozdělenou třetí vlnou ukazuje obrázek 2.2.

Selhání

Selhání u impulsu nastává, když pátá vlna nepřekročí konec třetí vlny.

Závěrečný impuls

Závěrečný impuls je takový impuls, kdy se cenový interval čtvrté vlny překrývá s cenovým intervalem vlny první. Závěrečný impuls signalizuje silnou změnu trhu.

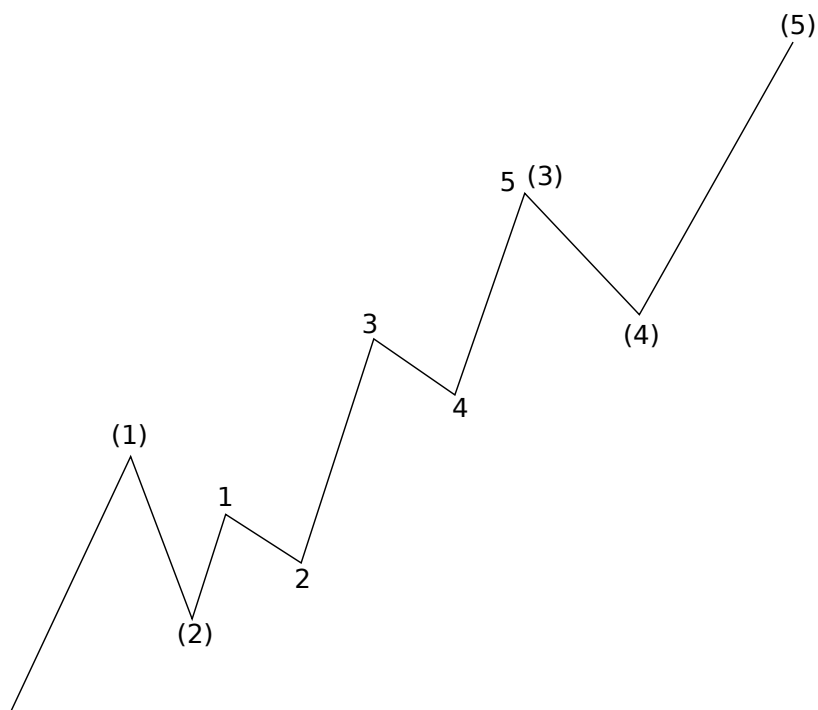
2.2.4 Korekce

Korekce se vyskytují mezi dvěma impulsy. Jsou tvořeny 1 nebo 3 vlnami nižšího stupně, jsou složitější a méně jednoznačné než impulsy avšak který celek není impulsem, musí být korekcí. Jednotlivé vlny korekce se značí písmeny. Korekce se dělí na několik typů: roviny, cikcak a trojúhelníky. Podle [6] dále korekce nemají výrazné poměry výšek mezi jednotlivými vlnami, ale výrazně se projevuje poměr zlatého řezu v délkách korekcí.

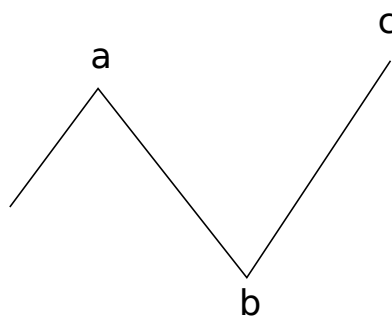
Rovina

Rovina se skládá ze tří vln: 2 korekcí a jednoho impulsu. Vlna b je vysoká alespoň 61,8% vlny a, vlna c alespoň 38,2% vlny a. Některé zdroje ([8], [7]) mají na rovinu vyšší pořadavky: vlna b je stejně velká jako a. Viz obrázek 2.3

Obrázek 2.2: Idealizovaný průběh hybného impulsu, třetí vlna je prodloužena a podrozdělena



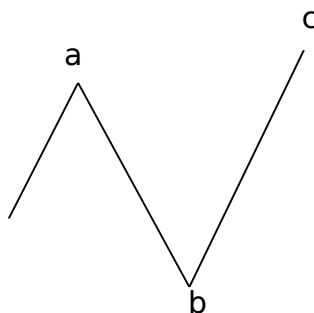
Obrázek 2.3: Idealizovaný průběh roviny



Nepravidelná

Podle některých zdrojů ([8], [7]) další forma korekce. Je podobná rovině, ale vlna b je větší než a. viz 2.4.

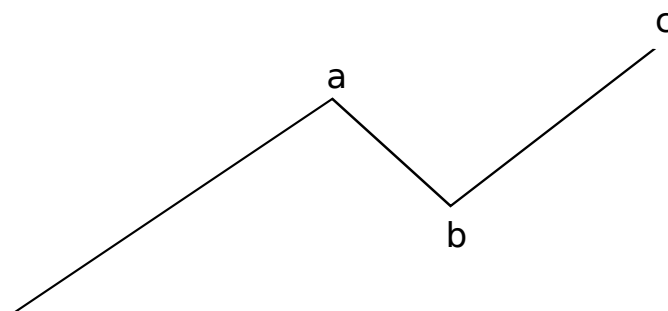
Obrázek 2.4: Idealizovaný průběh nepravidelné korekce



Cikcak

Skládá se z impulsu, korekce a impulsu. Vlna b by neměla být vyšší než 61,8% vlny a a vlna c by měla být vyšší než vlna b. Viz obrázek 2.5

Obrázek 2.5: Idealizovaný průběh cikcak



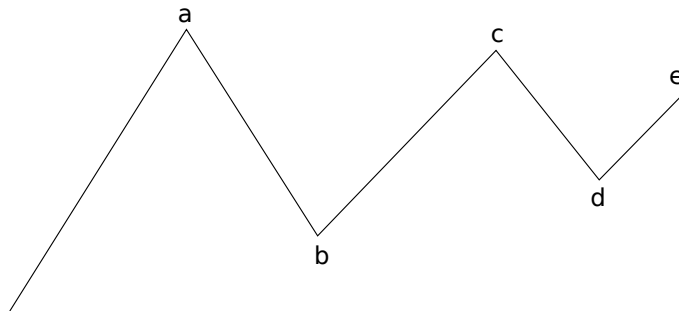
Trojúhelník

Trojúhelník se skládá pouze z korekčních vln, skládá se z 5 částí. Všechny části oscilují v určitém pásmu, které může mírně růst, nebo klesat. Trojúhelníky mohou být rozšiřující se nebo zužující se, kde každá následující část je vyšší, resp. nižší než předchozí. Rozlišuje se horizontální, rostoucí a klesající trojúhelník. Viz obrázek 2.6

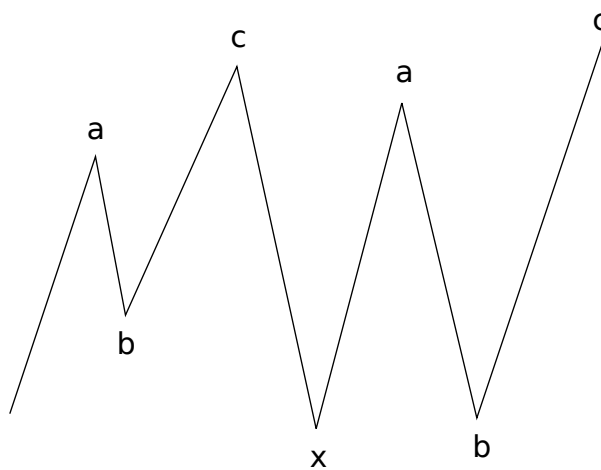
Dvojitá a trojitá trojka

Dvojitá, resp. trojitá trojka se skládá ze dvou, resp. tří korekčních vln se třemi podvlnami, které jsou spojeny x-vlnou. viz 2.7.

Obrázek 2.6: Idealizovaný průběh horizontálního zužujícího se trojúhelníka



Obrázek 2.7: Idealizovaný průběh dvojité trojky



2.3 Fibonacciho řada, zlatý řez a Elliottovy vlny

Bylo empiricky zjištěno, že délky a výšky po sobě následujících Elliottových vln často zachovávají určité poměry, které odpovídají po sobě jdoucím členům fibonacciho posloupnosti. Také počty vln nižšího stupně ve vlnách vyššího stupně jsou často fibonacciho čísla. Podle [6] pak nejčastější poměry délek, resp. výšek blízkých vln jsou 1.618 a 2.618.

Korekční vlny by měly dosahovat výšky 61, 8% vlny předchozí, v případě mírných korekcí 38, 2% výšky předchozí vlny.

2.3.1 Poměry výšek vln v impulsu

Poměry výšek hybných vln v impulsu by měly být v poměru Fibonacciho matematiky: 1:1, 1:1,618 a 1:2,618. V impulsu bez prodloužených vln by čtvrtá vlna měla dělit výšku impulsu ve zlatém řezu, 0,618 ku 0,382 ve prospěch první části. V případě, že je první nebo pátá vlna prodloužena, měla by prodloužená vlna být větší částí zlatého řezu. V případě, že je prodloužena třetí vlna, první a pátá vlna by si měly být rovny nebo v poměru do 1:1,618.

2.3.2 Poměry výšek vln v korekci

Ve vlně cikcak by si měly být výšky vln a a c rovny, v klasické ploché korekci by jim měla být rovna i vlna b. V rozšířené ploché korekci může být vlna c i 1,618 násobek vlny a.

V trojúhelníku jsou zpravidla 2 střídavé vlny (a-c,b-d,c-e) k sobě v poměru 1:1,618 (nebo 1,618:0)

2.4 Stupně vln

V závislosti na délce vlny Elliott pojmenoval devět stupňů vln. Jsou to *Velký supercyklus*, *supercyklus*, *cyklus*, *primární*, *prostřední*, *malý*, *menší*, *nejmenší* a *minimální*. Vlny těchto stupňů jsou značeny různými řezy a dekoracemi písma.

| Stupeň vlny | Trvání vlny | impuls | korekce |
|-------------------|-----------------|----------------|---------------|
| Velký supercyklus | stovky let | ⓪, ①, ... | ⓐ, ⓑ, ... |
| Supercyklus | desítky let | (I), (II), ... | (a), (b), ... |
| Cyklus | několik let | I, II, ... | a, b, ... |
| Primární | měsíce až roky | ①, ②, ... | Ⓐ, Ⓑ, ... |
| Prostřední | týdny až měsíce | (1), (2), ... | (A), (B), ... |
| Malý | týdny | 1, 2, ... | A, B, ... |
| Menší | dny | i, ii, ... | a, b, ... |
| Nejmenší | hodiny | (i), (ii), ... | (a), (b), ... |
| Minimální | minuty | ⓪, ①, ... | ⓐ, ⓑ, ... |

2.5 Současné systémy pro detekci Elliottových vln

Nejznámějším současným systémem pro detekci Elliottových vln je Elwave. [9] Software je uzavřený a tak nelze zjistit jeho vnitřní strukturu, ale podle popisu jeho funkčnosti pravděpodobně pracuje na principu vytváření vln od monovln k větším celkům podle Elliottových pravidel a v zásadě podle postupu uvedeného v 4.1.

Kapitola 3

Neuronové sítě a jejich použití při klasifikaci

Klasifikace je poměrně široké téma moderní informatiky. Zaměřím se pouze na použití neuronových sítí.

Klasifikací se rozumí rozdělování objektů do tříd podle určitých kritérií. Klasifikátor by tedy měl na základě vstupních dat vyjádřit, do které ze zkoumaných skupin vstupní objekt patří.

3.1 Neuronové sítě

Tato podkapitola popisuje základy použití neuronových sítí a jejich základní dělení. Celá vychází z [5].

3.1.1 Neuronová síť

Umělá neuronová síť je jeden z výpočetních modelů inspirovaných biologií. Základem každé neuronové sítě je neuron, [5] udává, že nejčastěji je používán neuron typu McCulloch and Pitts, který bude dále popsán. Neuron se skládá ze 2 částí, síťové funkce a aktivační funkce. Síťová funkce udává, jak jsou upraveny vstupní hodnoty, její výstup je vstupem aktivační funkce. Síťovou funkcí může být lineární kombinace vážených vstupů, nebo jiný způsob získání 1 výstupu z více vstupů.

Výstup síťové funkce je vstupem aktivační funkce, jejíž výstup je přímo výstupem neuronu. Funkci neuronu se vstupy y_1 až y_n , síťovou funkcí u , aktivační funkcí f a výstupem a popisuje následující rovnice 3.1.

$$a = f(u(y_1, y_2, \dots, y_n)) \quad (3.1)$$

$$f(u) = \frac{1}{1 + e^{-u/T}} \quad (3.2)$$

$$u(y_1, y_2, \dots, y_n) = \sum_{j=1}^n w_j y_j + \theta \quad (3.3)$$

Rovnice 3.2 udává aktivační funkci neuronu, zde logistickou sigmoidu. Dále jsou často podle [5] používány hyperbolická tangenta, prahování, lineární funkce, inverzní tangenta

nebo gaussovská radiální bázová funkce. 3.3 pak popisuje síťovou funkci, zde je uvedena podle [5] nejběžněji používaný vážený součet. θ pak zastupuje práh.

Rozdělení neuronových sítí

Neurony mohou fungovat samostatně (perceptron) nebo být seskupeny v sítích různé topologie. Dělit je obecně lze na cyklickou a acyklickou. Nejznámější acyklickou topologií neuronové sítě je *vrstevnatá neuronová síť*, kdy jednotlivé neurony používají jako vstupy pouze výstupy předchozí vrstvy neuronů. Její výhoda spočívá ve snadné implementaci trénování, které ovšem nemusí vést k nejlepším výsledkům. Nejznámější vrstevnatou sítí je multilayer perceptron. Acyklické sítě se dále dělí podle způsobu trénování, typu aktivační funkce a topologie propojení neuronů.

Cyklické, také rekurentní sítě modelují paměť a mohou tak modelovat složitější, nelineární a hlavně dynamické systémy. Nejznámějším zástupcem je radial basis network. V [5] je řečeno, že je možné radial basis network aproximovat pomocí multilayer perceptronu a naopak, rozdílem je hlavně rychlost učení podle typu dat. Dále uvádí, že radial basis network obvykle ke správné funkci vyžadují více dat.

Multilayer perceptron

Multilayer perceptron je plně propojenou vrstevnatou dopřednou neuronovou sítí. Jednotlivé neuronu jsou uspořádány do vrstev, vstupem neuronu každé vrstvy jsou všechny výstupy neuronů předchozí vrstvy, u neuronů první vrstvy je vstupem vstupní signál, tato vrstva se nazývá vstupní. Výstupní vrstva (často pouze jeden neuron) podává výstup celé sítě okolí. Vrstvy mezi těmito dvěma vrstvami se nazývají skryté [4].

Pro použití multilayer perceptronu je nutné nastavit váhy síťové funkce 3.3 tak, aby výstup při určitém vstupu nabýval takové hodnoty, které potřebujeme. Lze pak výstup perceptronu použít k detekci určitého vzoru na vstupu, nebo ke klasifikaci vzoru. Existuje mnoho metod pro trénování multilayer perceptronu, základní je error back propagation [5].

Error back propagation pracuje tak, aby minimalizovalo chybu každé epochy učení na K vzorech:

$$E = \sum_{k=1}^K [e(k)]^2 \quad (3.4)$$

Nová vstupní váha w v epoše $t + 1$ neuronu i ve vrstvě L pro výstup neuronu j ve vrstvě $L - 1$ jsou dány (je počito značení umístění pomocí horních i dolních indexů)

$$w_{i,j}^L(t+1) = w_{i,j}^L(t) + \eta \sum_{k=1}^K \delta_i^L(k) z_j^{L-1}(k) + \mu [w_{i,j}^L(t) - w_{i,j}^L(t-1)] + \varepsilon_{i,j}^L(t) \quad (3.5)$$

kde η je koeficient setrvačnosti, ε koeficient šumu a z je značen výstup neuronu. Derivace chyby $\delta_i^L(k)$ se vypočítá jako

$$\delta_i^L(k) = f'(u_i^L(k)) \sum_{m=1}^M \delta_m^{L+1}(k) w_{m,i}^L \quad (3.6)$$

kde u_i^L je výstup síťové funkce i -tého neuronu L -té vrstvy a m je číslování neuronů $L+1$ vrstvy a f' derivace aktivační funkce příslušného neuronu. Odvození těchto vzorů se nachází například v [5].

Podle článku [1] je pro detekci a klasifikaci určitých úseků úseků v časové řadě možné použít multilayer perceptron, pokud máme dostatečné množství trénovacích dat a nástroj proti přetrénování. Pokud budou tyto nástroje zavedeny, detekce podle oken může mít oproti použití paměti v síti výhodu přesnějšího nalezení vzoru.

3.1.2 Committee machines

Protože neuronové sítě vždy pracují s určitou chybou klasifikace, byly vyvinuty techniky pro skládání výsledků více neuronových sítí, tzv. Committee machines, u kterých se předpokládá, že chyby klasifikace zmenší, nebo na ně alespoň upozorní. Nejjednoduššími způsoby zpracování výstupů neuronových sítí ve výboru je průměrování, hlasování, bagging a boosting.

Průměrování jednoduše sečte výstupy neuronových sítí trénovaných na stejných datech. Hlasování při klasifikaci vybírá nejčastěji zvolenou třídu. Bagging je varianta průměrování, resp. hlasování, kdy jsou jednotlivé neuronové sítě trénovány na různých datech. Boosting při trénování sítí zvýhodňuje vzorky klasifikované předchozími trénovanými sítěmi špatně.

Mezi složitější committee machines patří Mixture of experts. Mixture of experts obsahuje síť specializované na určitou část prostoru vstupních dat, při hlasování nebo průměrování jsou výstupy neuronových sítí trénovaných na datech bližších vstupním zvýhodněny. Existuje mnoho variant provedení mixture of experts lišící se funkcí, která rozhoduje o dělení vstupního prostoru pro jednotlivé expertní sítě a také hloubkou dělení na jednotlivé expertní sítě. Mixture of experts se používá především v odhadování rizik. Její hlavní výhodou je rychlé učení, schopnost pracovat na velkých množinách vstupních dat a také vyšší přesnost.

3.2 Užití neuronových sítí pro klasifikaci

Neuronové sítě jsou hojně využívány pro detekci a klasifikaci v případě, že je známo množství klasifikovaných vzorků a je třeba určitého zobecnění.

Podle [4] byly v oblasti klasifikace neuronové sítě úspěšně použity pro řídicí systémy letadel, detektory závad komponent letadel, hodnocení rizikovitosti dlužníků, pro zaměřování zbraní, analýzu EKG a EEG, rozpoznávání hlasu, klasifikaci hlásek a jiné.

Kapitola 4

Možnosti detekce Elliottových vln

Elliottovy vlny s poměrně širokým popisem svých charakteristik mohou být poměrně obtížně detekovatelné přímo. Nejpřímějším přístupem k jejich detekci by bylo najít veškeré monovlny ve vstupních datech a na nich pak hledat části splňující kritéria pro jednotlivé Elliottovy figury. Tímto přístupem se zabývá 4.1.

Jinou možností je detekovat určité charakteristické znaky vln nejnižšího stupně a z těchto částí vytvářet jednotlivé figury. Teto přístup popisuje 4.2

Další a poslední zvažovanou možností je detekce přímo jednotlivých figur na základě jejich charakteristik, jak je popsáno v 4.3

4.1 Detekce na základě hierarchického systému pravidel

Jeden z možných přístupů k detekci a klasifikaci Elliottových vln je od nejmenších celků k největším. Tento přístup se používá při manuálním vyhodnocování časových řad a je podrobně zpracován. Popisují jej například Zdeněk Sojka a Petr Dostál v knize [6]. Vyhodnocování začíná nalezením měřítka a vyznačením monovln. Podle poměrů monovln jsou pak nalezeny možné figury a z těch sestavovány další. Úryvek z pravidel obsahuje podkapitola 4.1.1.

4.1.1 Pravidla pro identifikaci jednotlivých segmentů Elliottových figur

Pravidla pro identifikaci vln

Pro identifikaci vln používá [6] 7 pravidel, která určují vlny do jednotlivých skupin v závislosti na poměru výšek okolních vln. Pro účely přehlednosti je zkoumaná vlna značena m_1 a následující vlny postupně m_2 atd., předchozí vlna pak m_0 . Jednotlivá pravidla jsou číslována a vztahují se k poměru m_2/m_1 , dále se pak člení do podmínek a až f, které se vztahují k poměru m_0/m_1 . Jednotlivé poměry jsou odvozeny z Fibonacciho řady. Tabulku poměrů v procentech m_1 najdete v tabulce 4.1. Tyto poměry neplatí zcela přesně, je možná odchylka až 4%.

Pro popis jednotlivých pravidel budou použity strukturové štítky, které navíc určují pozici ve skupině vln. Budou to:

:F3 první vlna ve trojce (korekci)

:c3 prostřední vlna ve trojce

x:c3 střed trojky, spojovací vlna dvou korekčních figur

Tabulka 4.1: Pravidla přiřazená jednotlivým poměrům m_0/m_1 a m_2/m_1 (vyjádřeno v procentech) podle [6]

| | $m_0 < 38,2\%m_1$ | $38,2\%m_1 \leq m_0 < 61,8\%m_1$ | $61,8\%m_1 \leq m_0 < 100\%m_1$ | $100\%m_1 \leq m_0 < 161,8\%m_1$ | $161,8\%m_1 \leq m_0 < 261,8\%m_1$ | $261,8\%m_1 \leq m_0$ |
|---------------------------------------|-------------------|----------------------------------|---------------------------------|----------------------------------|------------------------------------|-----------------------|
| $m_2 < 38,2\%m_1$ | 1a | 1a | 1b | 1c | 1d | 1d |
| $38,2\%m_1 \leq m_2 < 61,8\%m_1$ | 2a | 2b | 2c | 2d | 2e | 2e |
| $m_2 = 61,8\%m_1$ | 3a | 3b | 3c | 3d | 3e | 3f |
| $61,8\%m_1 < m_2 < 100\%m_1$ | 4a | 4b | 4b | 4c | 4d | 4e |
| $100\%m_1 \leq m_2 < 161,8\%m_1$ | 5a | 5a | 5a | 5b | 5c | 5d |
| $161,8\%m_1 \leq m_2 \leq 261,8\%m_1$ | 6a | 6a | 6a | 6b | 6c | 6d |
| $261,8\%m_1 < m_2$ | 7a | 7a | 7a | 7b | 7c | 7d |

:sL3 předposlední ve trojce

:L3 poslední ve trojce

:5 pětka, impulsní formace

:s5 speciální pětka

:L5 poslední z pětky

Dále budou použity symboly pro zápis vlastností jednotlivých vln.

$t(mx)$ čas vlny mx

$h(mx)$ výška vlny mx (tj. rozdíl výšek začátku a konce vlny)

$p(mx) > p(py)$ při porovnávání vln stejného směru znamená, že vlna mx přesahuje vlnu my , tedy některá její část je dále ve směru pohybu vln

$v(mx)$ udává počet monovln, ze kterých se skládá mx .

$mx(:\mathbf{X})$ vlně mx je přiřazen strukturový štítek v závorkách

\approx zhruba odpovídá, zde znamená, že se velikost neliší o více než 61,8%.

\doteq je přibližně, tj. velikostní odchylka do 5%

Pravidlo 1a Převedení prvního pravidla popsaného v [6] do formulací výrokové logiky vypadá takto:

$$\begin{aligned}
& ((t(m2) \geq t(m1)) \vee (t(m2) \geq t(m3))) \Rightarrow m1(:5) \\
& ((t(m-1) \in \langle t(m0), 1, 681t(m0) \rangle) \wedge \\
& (h(m0) \doteq 0, 618h(m1)) \wedge (p(m4) < p(m0))) \Rightarrow m2(x:c3), m1(:s5) \\
& ((v(m0) > 3) \wedge (h(m1) > h(m0)) \wedge (t(m1) < t(m0))) \Rightarrow m0(\text{konec figury}) \\
& ((t(m0) \approx t(m2)) \wedge (h(m0) \approx h(m2)) \\
& \wedge (h(m(m-1)) \geq 1, 618h(m1)) \\
& \wedge (((h(m3) \geq h(m(-1)) \wedge (t(m3) \leq t(m(-1)))) \\
& \vee (h(\langle m3, m5 \rangle) \geq h(m(-1)) \wedge (t(\langle m3, m5 \rangle) \leq t(m(-1))))) \Rightarrow m1([:c3])
\end{aligned}$$

Ostatní pravidla mohou být popsána obdobně. Pro většinu kombinací vln $m-2$ až $m2$ s přihlédnutím k dalším vlnám existuje pravidlo, jaký štítek vlně přiřadit. Štítek se také často přiřazuje jiným vlnám než té, která je právě vyhodnocována. Jedné vlně tak bývá přiřazeno více štítků, jiné žádný. Pravidlo 1a patří mezi nejjednodušší, jaké tento klasifikační postup má.

Metoda je navržena k manuálnímu vyhodnocování. Předpokládá tedy určitou zkušenost hodnotitele a uvažování nad většími celky, než pouze vyhodnocovaným úsekem.

4.1.2 Provedení vyhledávání

Pravděpodobně jediná možnost vyhledávání je každou jednotlivou monovlnu spolu s jejím okolím prověřit kompletní hierarchií podmínek. Problémem pak ovšem zůstává sestavit z vln se štítky Elliottovy figury.

4.1.3 Hodnocení metody

Tato pravidla pro identifikaci monovln jsou sice jednoznačně popsána, avšak poněkud složitá. Jejich převedení pro použití při strojovém vyhodnocování je možné, pravděpodobně jej používají současné komerční detektory Elliottových vln.

Problémem metody je to, že buď můžeme detekovat pouze celky, které se skládají z přesně stanoveného počtu monovln, což bude velmi rychlé, ale mnoho figur nebude nalezeno, nebo umožnit skládání z většího počtu monovln, což povede k vysoké časové náročnosti algoritmu. Také není možnost, jak detekovat figury, které se skládají z menšího množství monovln, než mají dle pravidel mít.

Problém různě dlouhých monovln, resp. figur, které obsahují více monovln, než by měly, lze řešit vyhledáváním na figur v okně s omezeným množstvím monovln, spojováním monovln do možných vln jednotlivých figur, nebo hrubou výpočetní silou. Většina figur, které nemohou být detekovány pomocí monovln, protože jich díky vzorkování cen neobsahují dostatek, bude nalezena pomocí figur vyššího stupně.

4.2 Detekce charakteristických částí figur

Druhý možný přístup je nedetekovat na časové řadě monovlny, ale význačné formace, které jsou jednoznačné pro jednotlivé části různých Elliottových figur. Tyto formace by bylo

třeba nejdříve identifikovat na souboru potvrzených Elliottových figur. Vhodný vyhledávací prostředek, nejspíše neuronové sítě, by pak tyto formace vyhledával v časové řadě vstupních dat. Takto nalezené formace pak mohou být zpracovány na základě dalších parametrů, a to opět neuronovými sítěmi, nebo na základě přímého rozdělení, popřípadě kombinací obou.

4.2.1 Vytváření figur z charakteristických částí pomocí rozhodovacího stromu

Pro tento přístup k detekci Elliottových figur je nutné zvolit charakteristické části figur. Může to být formace vzestup ceny a menší pokles, vzestup ceny a větší pokles, růst a stejný pokles, přímý růst ceny, stagnace, všechny předchozí varianty pro pokles, konkávní a konvexní varianty a tak dále. Pak je z charakteristických vln stavěna na základě návaznosti jednotlivých částí Elliottova figura.

Elliottovy figury, tedy jednotlivé typy impulsů a korekcí by pak měly přiřazeny a ohodnoceny všechny přípustné průběhy těchto charakteristických částí a jejich jednotlivých poměrů délek a výšek, dále případně charakteristické části před a za figurou a mnoho dalších. S pomocí pravidel pak lze z nalezených charakteristických částí nalézt nejpravděpodobnější možnou sadu Elliottových figur na vstupních datech.

Toto vyhodnocování bude složité, se snižováním množství detekovaných charakteristických částí se může zjednodušovat, ale bude klesat jednoznačnost klasifikace figury.

4.2.2 Vytváření figur z charakteristických částí pomocí comittee machine

Pokud je k dispozici dostatečné množství testovacích dat, lze vyhodnocování nalezených charakteristických částí vedoucí k detekci a klasifikaci Elliottových figur provést pomocí metod boosting nebo pomocí Bayesovské komise. Díky velkému množství nalezených charakteristických částí může být detekce i klasifikace velmi přesná, avšak také pomalá.

K výhodám tohoto přístupu patří zejména možnost přesnější klasifikace, zejména oproti skládání monovln na základě systému pravidel, nalezení figur všech velikostí, už od několika vzorků a také určitá flexibilita ve vyhodnocování. Určitým problémem může být volba detekovaných charakteristických částí a dále nutnost velmi kvalitních trénovacích dat.

4.3 Detekce větších celků a jejich postupné dělení

Další přístup k detekci a klasifikaci Elliottových vln v grafu je postup od větších celků k menším. Vzhledem k poměrně jednoznačným podmínkám pro impuls jej lze detekovat oproti korekcím mnohem snadněji.

Impuls se od korekce liší především jasným určením směru a dále přesným popisem, jaká je velikost jednotlivých vln. Vytváří tak poměrně jednoznačný vzor, který může být poměrně snadno rozeznatelný. Je potom možné vytvořit neuronové sítě, které budou přímo detekovat přítomnost impulsu na vstupních datech.

Korekce, které jsou díky své mnohotvarosti přímo detekovatelné obtížněji, mohou být pak nalezeny pomocí základních pravidel Elliottových vln: Co není impulsem, je korekcí a korekce směřuje proti směru vlny vyššího stupně. Korekce tedy budou doplněny vždy mezi detekované impulsy stejného stupně a dále po nalezení vln v impulsích i do sudých vln v impulsích. Určitým problémem ovšem bude klasifikace korekcí, kdy bude o klasifikované korekci známo jen minimum informací.

Mezi hlavní výhody bude patřit určitá přímota detekce impulsů, kdy o impulsu bude známa jeho velikost a pozice již po první fázi detekce, dále nižší složitost algoritmu a odpovídající rychlost vyhledávání a také možnost detekce impulsů i na nejkratších úsecích dat. Hlavní nevýhoda již byly zmíněna: přímo jsou detekovány jen impulsy, korekce musí být odvozeny. To způsobí, že výstup se bude jevit více jednoznačný, než ve skutečnosti je. Další nevýhodou je to, že vnitřní struktura impulsů i odvozených korekcí je neznámá, pokud její součástí nejsou další detekované impulsy. Určitým problémem může být trénování neuronových sítí, kdy impuls je poměrně složitý útvar a může tedy dojít k přetrénování sítě nebo jejímu nedostatečnému natrénování.

4.4 Závěr

Vzhledem k nevýhodám detekce přímo z Elliottových pravidel (obtížná algoritmizace, mnohoznačnost pravidel) tuto metodu dále neuvažuji. Detekce Elliottových figur pomocí neuronových sítí detekuje pouze impulsy, což není zcela optimální. Z tohoto pohledu lepších výsledků dosahuje detekce charakteristických částí. Zde je hlavní nevýhodou nutnost velkého množství kvalitních trénovacích dat.

Kapitola 5

Detekční systém

5.1 Návrh systému

Program pro detekci Elliottových vln se bude skládat z několika částí. Základním blokem bude committee machine. Do budou vstoupovat data ze vstupních a normalizačních funkcí, výstupem committee machine pak bude množina nalezených Elliottových vln, u kterých committee machine určila vyšší než hraniční výsledek pro určitou třídu vlny.

Množina vln bude vstupem modulu postprocessingu, kde bude podle Elliottových pravidel vytvořena hierarchie vln (několik různých hierarchií vln, protože committee machine pravděpodobně nevytvoří pouze takové vlny, ze kterých půjde přímo vytvořit právě jedna hierarchie vln. Tyto hierarchie pak budou hodnoceny, vybrána nejlepší a podle té pak budou klasifikovány ostatní segmenty ve zkoumaném intervalu.

Tato hierarchie bude poté vykreslena nebo uložena pomocí výstupního modulu. Výstupem vstupně-výstupního modulu jsou tedy grafy Elliottových vln, dále soubory s jejich reprezentací.

Vstup do committee machine, tedy časovou řadu dat, je nutné načíst a normalizovat. To provede vstupní modul. Dalším vstupem committee machine je množina trénovaných neuronových sítí, kterou poskytne trénovací modul. Vzhledem k nedostatku trénovacích dat z praxe je nutné trénovací data generovat pomocí modulu generátoru.

Poslední důležitou částí programu je uživatelské rozhraní, které by mělo umožnit načíst časovou řadu, zadat její vyhodnocení, ovlivnit parametry vykreslení nalezených vln a také uložit výsledky klasifikace k dalšímu použití a také tyto výsledky načíst.

5.1.1 Trénovací modul a generátor Elliottových vln

Trénovací modul se skládá z funkcí, které použijí metody třídy multilayer perceptron a další funkce v modulu perceptronu k vytvoření a natrénování množiny sítí pro použití v committee machine.

Generátor Elliottových vln vytváří podklady pro učení sítí. Vzhledem k tomu, že jsem nedokázal získat dostatečné množství správně klasifikovaných Elliottových figur, bylo nutné vytvořit funkce, které budou Elliottovy figury generovat. Jelikož jedinou jednoznačnou figurou, jež lze snadno generovat je impuls (má přesně vymezený počet podvln, podvlny mají danou velikost), rozhodl jsem se, že přímou klasifikaci vln omezím na detekci impulsů a generátor bude vytvářet pouze impulsy a neimpulsy, tedy náhodné vlny, které nejsou impulsem.

Elliottovy vlny mají určité poměry výšek jednotlivých vln. Dále pro impuls platí pravidla, uvedená například v 2.2.3. Posloupnost šesti bodů s časovou a cenovou souřadnicí udávající zlomové body impulsu je tedy sestavena na základě pseudonáhodného výběru z možných platných podvln. Tato posloupnost je poté otestována, zda splňuje podmínky kladené na impuls a uložena k dalšímu použití. Na základě časového pravidla a zlatého řezu je vytvářena také pseudonáhodná posloupnost bodů, které nemohou být impulsem. Generátor může vytvářet také posloupnosti, které jsou částmi impulsů nebo korekcí.

Učení sítě probíhá s učitelem. Pro učení sítí jsou použity generované impulsy a neimpulsy k učení jednotlivých sítí. Každá síť je trénována na jiných datech, aby se jejich výsledky více lišily a výsledky baggingu v committee machine byly lepší. Trénovací data jsou vzorkována na požadovaný počet vstupů a jejich dávka je vložena do učícího algoritmu sítě. Váhy síťové funkce jsou upraveny podle výsledků učícího algoritmu sítě a zaznamenány. Při další epoše učení je zaznamenána chyba klasifikace, která patří váhám vytvořeným předchozím učením a je jim přiřazena, dále nové váhy, které jsou uloženy k vyhodnocení v další epoše učení. Z takto porožených vah jsou po ukončení učení vybrány a uloženy váhy s nejnižší naměřenou chybou.

Učení řídí jednoduchý konečný automat podle vývoje chyby. Pokud chyba neklesá, jsou upraveny parametry učící procedury sítě směrem k větší náhodnosti, pokud chyba neklesá delší počet epoch, je učení ukončeno.

5.1.2 Multilayer Perceptron

Vzhledem k tomu, že během tvorby práce neexistovala implementace neuronové sítě v jazyku python 3 jsem se rozhodl vytvořit implementaci vlastní. Z důvodu jednoduchosti a účinnosti jsem se rozhodl pro multilayer perceptron, dále MLP. Bude založen na vrstvách neuronů dle McCullocha a Pittse, síť bude dopředná, vrstevnatá a plně propojená.

Základem je třída MLP, která obsahuje vstupní váhy jednotlivých vrstev, aktivační funkce jednotlivých vrstev a jejich derivace pro učící algoritmy, názvy klasifikovaných tříd a informaci o tvaru sítě. Poskytuje metody pro vytvoření sítě, její uložení, průchod dat sítí a průchod sítí pro učení, který vrací i mezivýsledky jednotlivých neuronů v síti. Síť je možné zvolit libovolného tvaru, výběr aktivačních funkcí je omezen na jednu funkci na vrstvu sítě.

Modul Perceptronu dále musí obsahovat základní aktivační funkce a jejich derivace, a především proceduru pro učení sítě. Nezbytnou aktivační funkcí bude logistická sigmoida a dále lineární funkce.

Pro učení sítě byl zvolen a implementován algoritmus Error back propagation se setrvačností a náhodným šumem, popsáný v 3.1.1.

5.1.3 Vstup a preprocessing

Vstupem do systému bude časová řada uložená v souboru. Preprocessing ji rozdělí pomocí plovoucích oken na řady segmentů různé délky. Dále vytvoří segmenty, které obsahují každý n-tý prvek vstupních dat, aby mohly být detekovány i velmi dlouhé impulsy.

Následně procesura preprocessingu jednotlivé segmenty normalizuje podle typu MLP, do kterého budou vloženy. Vzhledem k tomu, že trénovací data byla uměle vytvořena, preprocessing neobsahuje žádné kroky, které by měnily velikost vstupního vektoru protože na umělých datech by mohla být zvolena špatná forma úpravy dat.

Preprocessing tedy sestává buď z normalizace vstupního segmentu tak, aby veškeré hodnoty ležely v intervalu $(0, 1)$, nebo pouhého odečtení sousedních hodnot a následné úpravy

tak, aby bylo zachováno znaménko každé hodny a hodnoty ležely v intervalu $\langle -1, 1 \rangle$. Jednotlivé sítě jsou učeny na datech s jednou z těchto úprav, aby se dosáhlo větší různorodosti sítí.

5.1.4 Committee machine

Committee machine bude pracovat s několika sadami MLP učenými na různých datech, s různým tvarem sítě a různým předzpracováním vstupu, čímž bude zajištěna různost sítí pro lepší funkci porovnávání výstupních dat. Jelikož jsou MLP trénovány na detekci rostoucího impulsu, průchodem všech dat přes množinu MLP a následným průchodem záporných dat přes množinu MLP získáme velké množství výstupů, které dále budou interpretovány jako forma vyjádření pravděpodobnosti výskytu rostoucího impulsu začínajícího na počátku segmentu s délkou rovnou délce segmentu.

Protože výstupy MLP jsou normalizovány na interval $\langle 0, 1 \rangle$, lze aritmetickým průměrem výstupů sítí nad kladnými daty a odečtením průměru výstupů sítí nad zápornými daty určit číslo p , které udává směr pravděpodobného impulsu na prohledávaném segmentu a podle velikosti čísla i určitou formu pravděpodobnosti. Segmenty, u kterých toto číslo přesáhne určitou mez pak budou prohlášeny za impulsy s pravděpodobností výskytu p a předány k dalšímu vyhodnocení. Ostatní vyhodnocení jsou zahozeny.

Jednotlivé segmenty budou ukládány jako objekty třídy `impuls`, která je podtřídou třídy `segment`. Tato třída bude obsahovat odkaz na data, na kterých byl segment detekován, dále seznam jednotlivých segmentů o jedna nižšího stupně, ze kterých se daný segment skládá, index začátku a konce segmentu na vyhodnocovaných datech, a dále metody pro určení statistických dat o segmentu. Committee machine vytvořenému segmentu typu `impuls` dále přiřadí parametr pravděpodobnosti p . Pro ilustraci vyhledávání impulsů jsou impulsy nalezené pomocí committee machine během testování programu uvedeny na obrázku 5.1.

5.1.5 Tvorba hierarchie

Tvorba hierarchie vln bude mít za úkol z jednoduchých impulsů nalezených pomocí committee machine vytvořit sady hierarchií, u těch vyhodnotit shodu s předpoklady o Elliottových vlnách a vybrat takovou hierarchii vlny, která tyto požadavky splní nejlépe.

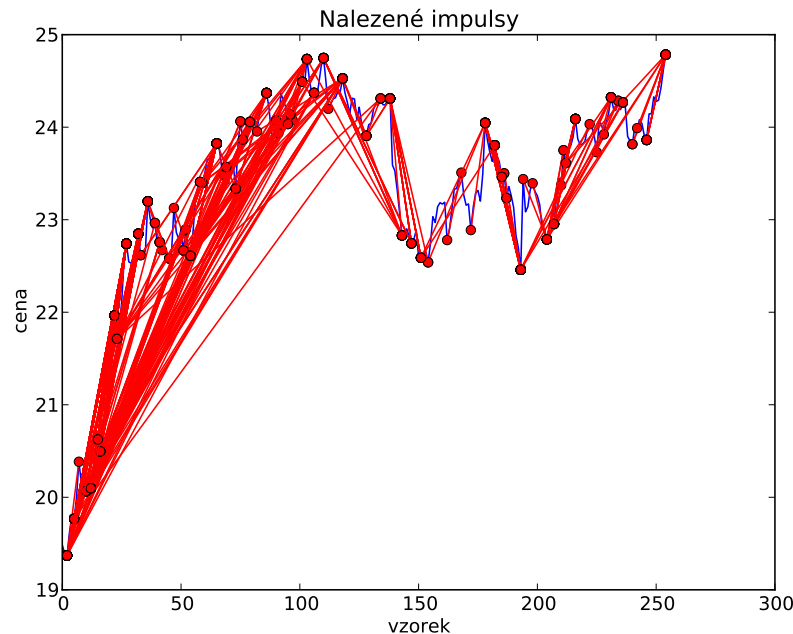
Prvním krokem při tvorbě hierarchií bude upravit nalezené impulsy tak, aby odpovídaly zlomům v datech. Tím se zmenší množství impulsů, se kterými je nutné pracovat, protože committee machine nevyhodnocuje, jestli na segmentech blízkých analyzovanému již nebyl nějaký impuls nalezen. Proto prvním krokem při tvorbě hierarchie musí být nalezení monotonních úseků na analyzovaných datech, tyto úseky budou označeny za monovlny.

Následně budou impulsy nalezené pomocí committee machine upraveny tak, aby obsahovaly pouze celé monovlny. Vzhledem k možné chybě při určování monovln náležících do impulsu je vytvořeno několik nových impulsů s různým počtem monovln.

Protože počet impulsů je po předchozím kroku příliš velký, je nutné vyřadit takové impulsy, z jejichž monovln nelze sestavit podvlny impulsu vyhovující pravidlům Elliottových vln. Impulsy, které nevyhovují pravidlům, protože mají příliš málo monovln budou uloženy k použití při vytváření hierarchie, ostatní nevyhovující impulsy budou zahozeny. Pro ilustraci průběhu vytváření hierarchie vln je uveden obrázek 5.2. Do impulsů jsou vloženy seskupení vln, které vyhovují pravidlům Elliottových vln.

Z impulsů potvrzených v předchozím kroku je nyní vytvořena hierarchie. V jednotlivých sadách seznamů podvln impulsu budou podvlny, jejichž délka a začátek odpovídá některému z potvrzených impulsů nebo některému z impulsů příliš krátkých na to, aby mohl být

Obrázek 5.1: Impulsy nalezené pomocí committee machine, impulsy jsou značeny červenými úsečkami se zvýrazněným začátkem a koncem. I přes nepřehlednost si lze všimnout, že se mnoho impulsů, začíná na podobném, ale nikoli stejném místě. Množství impulsů je důvodem pro filtrování impulsů před vytvářením hierarchické struktury.



potvrzen, nahrazeny tímto impulsem. Tím vznikne několik hierarchických struktur, které sdílí společné prvky.

Každý z impulsů, který obsahuje podvlny, pak musí být znovu hodnocen. Každý ze seznamů podvln je hodnocen podle impulsů, které obsahuje, dále podle poměrů velikostí jednotlivých podvln a míry vybočení jednotlivých podvln z ideálního tvaru. Pak je vybrán nejlépe hodnocený seznam, podvlny z tohoto seznamu jsou přiřazeny impulsu a impuls přebírá hodnocení nejlepšího seznamu podvln.

Impulsy pak jsou zařazeny do hierarchie tak, aby vzájemně nekolidovaly. Provedení je možné jednoduchým algoritmem, zde v pseudokódu. Algoritmus postupně zařazuje do hierarchie vln vlny ze seznamu impulsů s , kde jsou impulsy seřazeny podle hodnocení podvln:

opakuj:

 procházej členy s :

 pokud se člen nepřekrývá:

 přidej ho do hierarchie a skonči cyklus procházení

 pokud se člen překrývá s jinými členy hierarchie:

 pokud jsou všechny podvlnami přidaného členu:

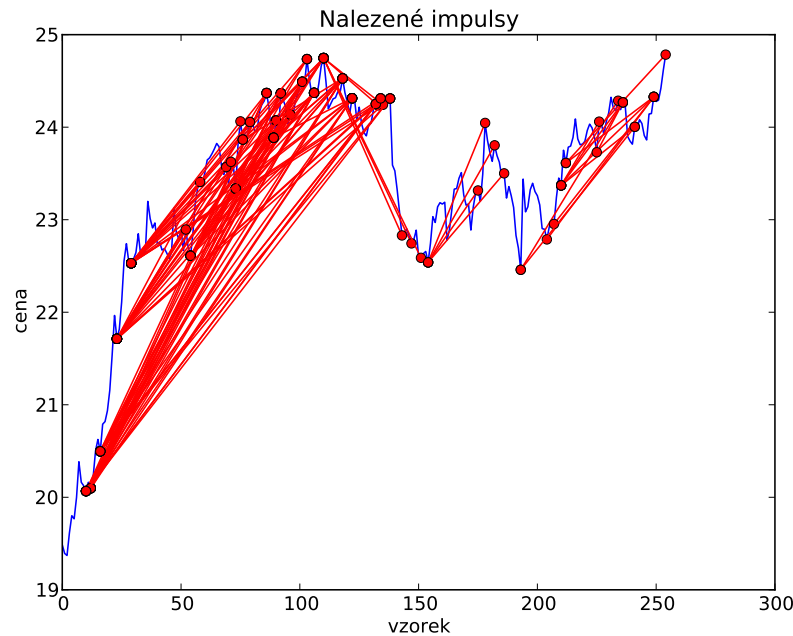
 přidej nový člen a odstraň překrývající se z hierarchie

 dokud je přidán během cyklu člen

Tak je vytvořena hierarchie obsahující všechny nejlépe hodnocené impulsy.

Podle nalezené hierarchie a pravidel Elliottových vln je možné dosud neklasifikované

Obrázek 5.2: Impulsy, v nichž lze sestavit 5 platných segmentů. Jde o výstup ze stejného vyhodnocování jako 5.1, zde jsou již impulsy pročištěny od impulsů, které jsou velmi podobné jiným. Lze si všimnout, že impulsy začínají pouze v několika bodech a liší se délkou, což bude důležité pro další tvorbu hierarchie vln



podvlny impulsů klasifikovat jako impulsy nebo korekce pomocí funkcí pro vyhodnocování podvln a tak vytvořit kompletní hierarchii vln. Pro ilustraci je uvedena hierarchie Elliottových vln vytvořená na stejných datech jako předchozí obrázky 5.3.

Poslední bod vytváření hierarchie, tedy z největších nalezených impulsů skládat nadvlny, bude součástí dalšího rozvoje programu.

5.1.6 Výstupní modul

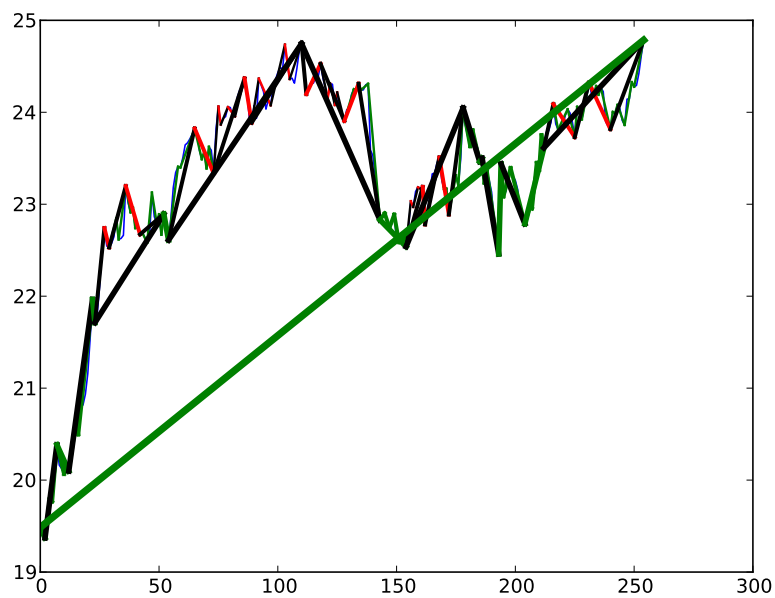
Výstupní modul vytvoří z nalezené hierarchie vln výstup čitelný člověkem. Výstupem by měl být graf nalezených vln s možností volby, jaká hloubka hierarchie vln má být zobrazena. Dalším vhodným výstupem by byla data o nalezených vlnách formě pro další zpracování. Pro funkce programu bude vhodné mít k dispozici jednoduché ukládání nalezené hierarchie pro budoucí načtení a práci s nalezenými vlnami.

5.1.7 Uživatelské rozhraní

Uživatelské rozhraní zprostředkovává funkce programu okolí. Vzhledem k značně pomalé detekci je vhodné rozhraní vytvořit tak, aby bylo použitelné pro dávkové zpracování. Program tedy musí být schopen pracovat z příkazové řádky.

Dále by bylo vhodné, aby program obsahoval určité interaktivní funkce pro řízení vykreslování grafů.

Obrázek 5.3: Nalezená hierarchie impulsů. Impulsy jsou označeny černými úsečkami, korekce červenými, neklasifikované segmenty zelenými. Jsou zobrazeny segmenty do třetí úrovně podvl.



5.2 Možnosti ovlivnění funkce systému

Funkce systému je možné ovlivnit mnoha způsoby, většinou však přímo v kódu skriptů.

5.2.1 Perceptrony

Nejjednodušší metodou ovlivnění chodu programu je přidání, popřípadě odebrání detekčních neuronových sítí. Program hledá neuronové sítě v adresáři data, za neuronovou síť považuje soubory **.mlp*. Pro vyřazení určité sítě z vyhodnocování stačí pozměnit tuto příponu.

Pro vytváření nových sítí slouží procedura *makeMLPs* v modulu Trainer, která vytvoří žádanou sadu sítí se základním pojmenováním.

Úpravou konstant ve funkcích *trainMLPn* a *trainMLPd* lze měnit parametry učení sítí, konstanty uložené ve výchozím stavu vedly k nejlepším výsledkům.

5.2.2 Parametry filtrování impulsů v committee machine

V committee machine je vhodné měnit konstanty pouze u vyhledávání impulsů v proceduře *findImpulses* a to pouze konstanty pro limity pravděpodobnosti pro detekci impulsu.

5.2.3 Parametry tvorby hierarchie

Při tvorbě hierarchie se používá mnoho konstant, které byly empiricky zjištěny během vývoje jako nejvhodnější pro detekci. Tyto konstanty se nachází v procedurách *testFibonacci*, *testSubimpulses* a *testStandardWaves* které hodnotí možné podvlny impulsu.

Kapitola 6

Výsledky

Výsledkem práce je program, který nachází Elliottovy vlny v zadané časové řadě. Program lze ovládat z příkazové řádky nebo textovými příkazy v příkazové řádce programu. Program je sadou skriptů psaných v pythonu 3.2, což je také minimální verze pythonu nutná pro správný běh programu. K běhu programu je dále třeba knihovna numpy verze alespoň 1.5 a knihovna matplotlib verze 1.0.

Shrnutí výstupů programu obsahují následující podkapitoly. Poznámka: osy nejsou popsány úmyslně, program pracuje s bezrozměrnými daty a není tedy důvod uměle přidávat rozměr. Horizontální směr je vždy časem a vertikální cenou.

6.1 Detekce testovacích impulsů pomocí různých sad sítí

Při detekci impulsů obklopených náhodnými daty si detekční systém vede poměrně dobře, viz obrázek detekovaných vln pomocí plné sady sítí [6.1](#).

Výsledky se příliš neliší ani při zmenšování množiny neuronových sítí, a to až po nejmenší funkční sadu 12 sítí.

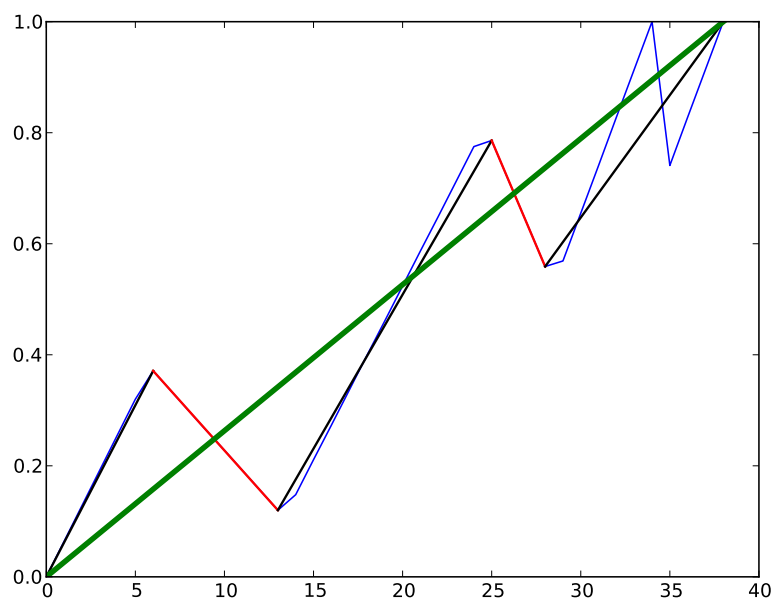
6.2 Detekce impulsů na časových řadách finančního trhu

Pro detekci byla použita skutečná data finančního trhu, konkrétně pak kurzy koruny vůči světovým měnám za rok 1999 podle údajů České národní banky.

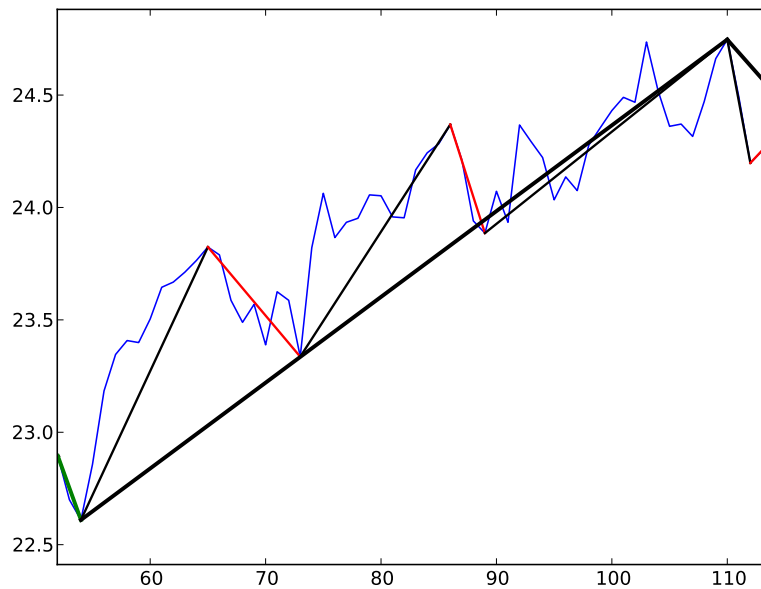
Z grafů [6.2](#) a [6.3](#) je zřejmé, že přílišné omezení množiny neuronových sítí může vést k ne zcela správné detekci jednotlivých vln impulsu, protože nejkratší vlny nejsou detekovány správně. Zrychlení programu je spíše neznamenné, většinu času totiž trvá zařazování impulsů do hierarchie vln.

Naopak u [6.4](#) a [6.5](#) dopadla detekce stejně. Je sice sporné, zda by na vlny identifikoval profesionál v oboru stejně, ale základní poučky Elliottovy teorie splňují.

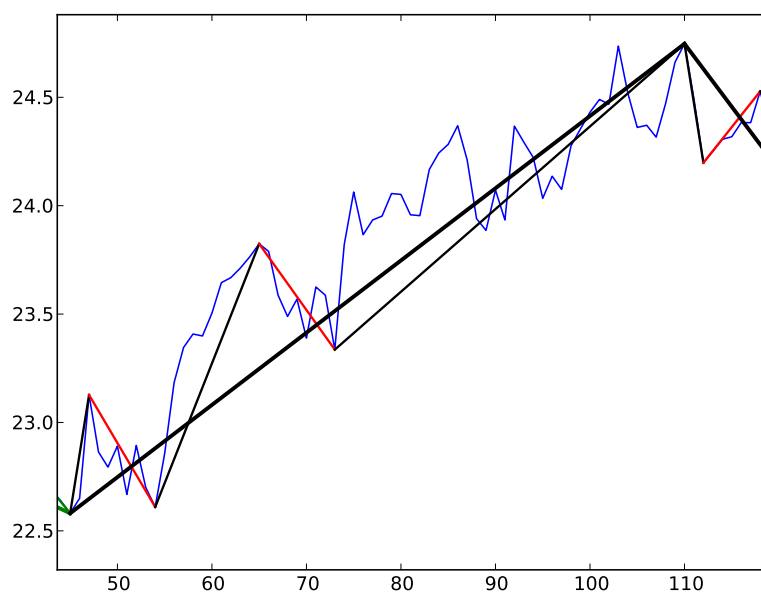
Obrázek 6.1: Nalezená hierarchie impulsů na generovaném impulsu při úplné sadě neuronových sítí. Impulsy i korekce jsou nalezeny správně. Impulsy jsou označeny černými úsečkami, korekce červenými, neklasifikované segmenty zelenými. Jsou zobrazeny segmenty do třetí úrovně podvln.



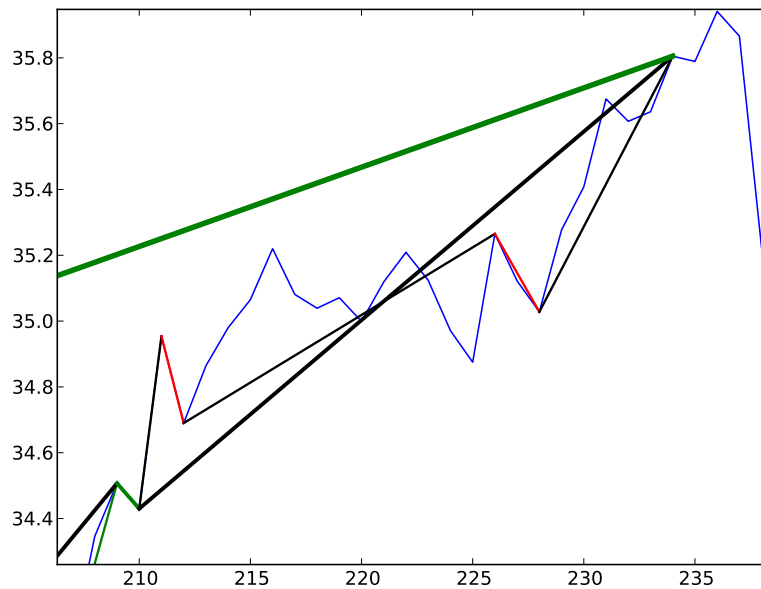
Obrázek 6.2: Výřez z hierarchie impulsů na datech kanadského Dolaru, detekce pomocí plné sady neuronových sítí



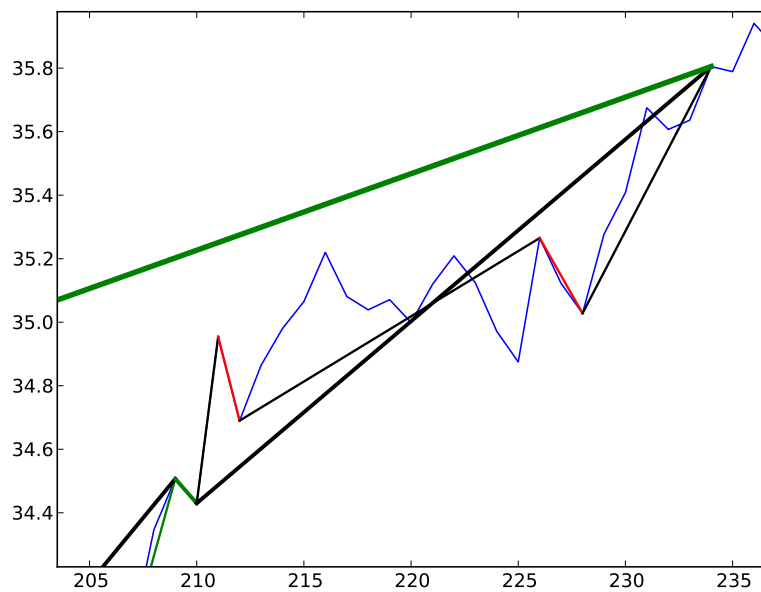
Obrázek 6.3: Výřez z hierarchie impulsů na datech kanadského Dolaru, detekce pomocí omezené sady neuronových sítí



Obrázek 6.4: Výřez z hierarchie impulsů na datech Dolaru, detekce pomocí plné sady neuronových sítí



Obrázek 6.5: Výřez z hierarchie impulsů na datech Dolaru, detekce pomocí omezené sady neuronových sítí



Kapitola 7

Závěr

V rámci práce bylo navrženo a popsáno několik způsobů detekce Elliottových vln v časové řadě. Byl zvolen přístup, kdy je časová řada chápána jako statický obraz a jsou hledány vzory odpovídající impulsu pomocí committee machine používající multilayer perceptron. Výhodou tohoto postupu je přímé nalezení začátku a konce detekovaného impulsu, nevýhodou poněkud větší časová náročnost.

Protože nejsou k dispozici klasifikovaná trénovací data, byla vytvořena procedura generující impulsy použité k učení sítí.

Podle navržených algoritmů byl vytvořen program, který dokáže s určitou přesností najít v časové řadě Elliottovy vlny. Tento program sice nedosahuje zcela správných výsledků, ale pro určitou představu o dění na trhu výsledky mohou postačovat.

Největší slabinou programu je jeho rychlost, respektive pomalost. Zdaleka nejpomalejší částí je zařazování vln do hierarchie, které by mělo být přepracováno. Další částí, která program značně zpomaluje je průchod dat skrz committee machine. Ten může být značně urychlen zmenšením počtu použitých neuronových sítí, nebo přepsáním třídy perceptron tak, aby používala optimalizovaný kód pro práci s maticemi. Zmenšení množiny neuronových sítí však vede k horším výsledkům detekce nejkratších vln.

Další možnost rozvoje detekce vln doplnění vytváření hierarchie vln i směrem vzhůru, které tato práce neřeší. K plnohodnotnému systému pro detekci Elliottových vln také schází detekce kanálů v Elliottových vlnách a přesnější detekce a klasifikace korekcí.

Literatura

- [1] Dorffner, G.: Neural Networks for Time Series Processing. *Neural Network World*, 96.
- [2] Elliott, R. N.: *The Wave Principle*. 1938.
- [3] Frost, A. J.; Prechter, R. R.: *Elliott wave principle: key to market behavior*. New Classics Library, 1998, ISBN 978-0-932850-75-4.
- [4] Hagan, M. T.; Demuth, H. B.; Beale, M. H.: *Neural Network Design*. PWS, 1995, ISBN 0-9717321-0-8.
- [5] Hu, Y. H.; Hwang, J.-H.: *Handbook of neural network signal processing*. CRC Press, 2002, ISBN 0-8493-2359-2.
- [6] Sojka, Z.; Dostál, P.: *Elliottovy vlny*. Tribun EU, 2008, ISBN 978-80-7399-630-7.
- [7] WWW stránky: Acrotec. <http://www.acrotec.com/ewt.htm>.
- [8] WWW stránky: Ardeus - Online Trading Education. <http://www.ardeus.cz/komodity-online/TECHNICKA-ANALYZA/Elliottivy-vlny.html>.
- [9] WWW stránky: ELWAVE Elliott Wave Software. <http://www.prognosis.nl/elwave/>.

Příloha A

Manuál

A.1 Instalace a požadavky na funkci programu

Program k funkci vyžaduje interpret python 3.2 a novější s instalovanými knihovnami numpy a matplotlib. Instalace programu spočívá v přesunutí skriptů pythonu a složky data s obsahem do adresáře, kde je uživatel oprávněn zapisovat.

Kombinace python 3.2 a matplotlib zatím není oficiálně podporována, ale funkce nutné pro práci programu jsou funkční.

A.2 Ovládání programu v příkazové řádce

Argumenty programu v příkazové řádce jsou:

- h nebo help vytiskne nápovědu
- n nebo interactive spustí interní příkazový řádek programu
- i nebo input načte soubor ve formátu csv specifikovaný parametrem
- c nebo input_CNB načte soubor ve formátu, v jakém česká národní banka publikuje archivní data kurzů měn. Druhý parametr pak udává měnu, jejíž kurzová data se mají načíst. Udává se ve formátu jako v hlavičce tabulky, místo mezery je třeba použít podtržítko.
- d nebo detect provede detekci vln na načtené časové řadě.
- l nebo load načte soubor, do kterého byly uloženy již detekované vlny.
- s nebo save uloží soubor s vlnami k pozdějšímu načtení a zobrazení.
- e nebo export uloží do zadaného souboru textovou reprezentaci nalezených vln.
- f nebo figure_save uloží do zadaného souboru graf s vykreslením N zadaných úrovní. Parametr s počtem úrovní se zadává první. Podle přípony cílového souboru bude výkres graf uložen v daném formátu. Podporované formáty jsou eps, png, psd, raw a svg.
- v nebo view vykreslí graf s n zadanými úrovněmi.

A.3 Režim textových příkazů

Režim vnitřní příkazové řádky podporuje veškeré funkce jako spouštění programu a argumenty a neposkytuje žádné rozšíření funkčnosti. Jeho účelem je snadnější vykreslování a ukládání grafů již dříve detekovaných impulsů. Používá příkazy obdobné argumentům při spouštění v příkazové řádce.

A.4 Typické použití programu

```
user@system: python eliottfinder.py -i input.txt -d -s out.segm
data nahrána
...
potvreno 34
user@system: python eliottfinder.py -l out.segm
>view 2
>view 3
>print 3 out.png
>exit
```

Tento postup načte data časové řady ve formátu CSV, provede detekci a uloží výsledek do souboru out.segm. Při druhém spuštění je otevřen soubor out.segm, zobrazen graf pro 2 a následně tři vrstvy podvln, graf pro tři vrstvy podvln je uložen ve formátu png a program ukončen.

A.5 Požadavky na vstupní data

Na vstupní data ve formátu csv nejsou kladeny velké nároky. Musí obsahovat pouze čísel s desetinnou tečkou, jednotlivá čísla oddělena tabulátory, mezerami nebo čárkami. Na řádkování není brán zřetel.

A.6 Snímek obrazovky programu

Obrázek A.1: Snímek obrazovky po provedení detekce a zobrazení

