Budapesti Corvinus Egyetem

Közgazdaságtudományi Kar

Kochmeister pályázat

Portfoliókezelés adatbányászati eszközökkel

Készítette: **Badics Milán Csaba**

Közgazdaság elemző Msc II. évfolyam

Konzulens: Ferenczi Tamás

2014.05.02.

Tartalom

Köszönetnyilvánítás

Ezúton mondok köszönetet Ferenczi Tamásnak, aki segített a releváns szakirodalom megértésében, és a modellek programozásának kivitelezésében. Szintén köszönettel tartozom Hans Zoltánnak és Szoboszlai Mihálynak akik átolvasták dolgozatomat és hasznos tanácsokkal láttak el a kutatási eredményeim megfogalmazása során.

A dolgozatban maradt hibákért természetesen minden felelősség engem illet.

Bevezetés

A tőzsdei idősorok alakulása már évtizedek óta a befeketetők figyelmének középpontjában áll és próbálják különféle módszerekkel előrejelezni azt. A nagy érdeklősédre való tekintettel akadémai körökben is egyre több kutatás kezdett el foglalkozni a tőzsde idősorok előrejelzésének lehetőségeivel. Először a közgazdaságtanban használatos hagyományos statisztkai/ökonometriai modelleket kezdték el használni, azonban az idősorok speciális jellegzetességi mint például a nemlinearitás, nem-stacioner tulajdonság, magas zaj/jel arány miatt ezek kevésbé bizonyultak eredményesnek. Ekkor fordultak a műszaki életben gyakran alkalmazott nem-paraméteres, kevesebb statisztikai megkötéssel rendelkező adatbányászati/gépi tanulási módszerek felé, és ezek eszköztára új perspektívákat nyújtott az tőzsdei idősorok hatékonyabb előrejelzésére. Az elmúlt 30 évben az adatbányászati módszerek egyre több változatát kezdték el használni tőzsdei adatok alakulásának vizsgálatára. Először az egyik legnépszerűbb adatbányászati módszert a neurális hálók különböző fajtáit használták a statisztikai módszerekhez képest nagy előnnyel. Mivel az előrejelzési pontosság kicsi javítása is akár hatalmas többletprofitit eredményezhet ezért mind a befektetői mind az akadémai körökben egyre népszerűbb lett a különböző hálózatok közül a legoptimálisabb megkeresése, annak megfelelő parametrizálása. A közgazdászok helyett egyre inkább fizikusok, matematikusok és informatikusok kezdtek el foglalkozni az előrejelzés módszertanával köszönhetően annak hogy ezek a modellek jelentős elméleti és módszertani hátteret követelnek meg mind a fejlesztőktől mind az alkalmazóktól egyaránt. Azonban idővel a nagy sikerre és a széles körű alkalmazásra való tekintettel -mint ahogy minden előrejelzésre épített stratégia, ha sokan kezdik el használni egyszerre- az erre épített befektetési döntések átlagon felüli profitszerzési egyre csökkent. Ugyanakkor ez nem jelentette azt hogy a befektetési döntéshozók, illetve a kutatók ezután elutasították volna ezen módszerek használatát, éppen ellenkezőleg, egyre több és több eneregiát fektettek bele a műszaki élet egyéb területein már sikerrel használt módszerek idősor-előrejelzésre való átültetésére. Többek között elkezdték használni a többi adatbányászati módszer módosított változatait (SVR, Random Forest), illetve a zajszűrő (ICA, PCA) és dekompozíció alapú (EMD, wavelet) technikákat is. Emellett elterjedt a többlépcsős hybrid módszerek alkalmazása és az egyes előrejelzések kombinálása is. Mára már rengeteg módszer és modell került kifejlesztése így a tőzsdei idősor-előrejelzésen alapuló stratégia az egyik

legnépszerűbbek közé tartozik, ugyanakkor alkalmazása egyúttal komoly kihívás is, mivel a hatékony előrejelzéshez szükséges a különböző modell előnyeinek és hátrányainak ismerete.

Dolgozatomban ezért bemutatom a legismertebb aktív portfoliókezelésre alkalmas adatbányászati módszereket, azok előnyeit és hátrányait, melyiket mikor és milyen formában érdemes alkalmazni, illetve kitérek arra is, hogy mik a jelentősebb kutatási irányok napjainkban. Célom hogy a teljes folyamatot bemutassam az előrejelzeni kívánt részvényárfolyamok kiválasztásától (dolgozatomban az OTP és a MOL napi záróárfolyamai), a szükséges input változók és a használható adatbányászati módszerek definiálásán át egészen az optimális portfolió kialaktásáig, mintegy útikönyvet adva ezzel az olvasó kezébe az előrejelzésen alapuló aktív portfoliókezeléshez.

Dolgozatomat a tőzsdei idősorok jellemzőinek, és az elmúlt évtizedekben használt statisztikai és adatbányászat modellek bemutatásval kezdem. Itt térek ki arra, hogy milyen okok vezérelték a kutatókat utóbbi módszerek használatának elterjedésére, a tőzsdei idősorok milyen jellemzői indokolják használatukat. Eztuán röviden ismertetem az adatbányászati projektek módszertanát, hogyan épül fel egy álatalános adatbányászati kutatás, és hogy ez miben tér el pénzügyi idősorok előrejelzése esetén, milyen kihívásokkal kell szembenáznie ilyen esetekben modellezőnek. A következő fejezetben részeltesen bemutatom az egyes lépések esetén alkalmazható módszerek előnyeit hátrányait, majd a dolgozatomban ezek közül alkalmazásra kiválaszottak elméleti hátterét is. Végezetül a Budapesti Értéktőszde két részvényének adatait felhasználva az általam legjobbnak vélt adatbányászati módszerek felhasználásával megpróbálok a „buy and hold” stratégiánal egy eredményesebb, előrejelzésen alapú aktív portfoliókezelési stratégiát bemutatni. Dolgozatom végén kitérek majd a módszer nehézségeire, és a további lehetséges kutatási irányokra is.

Reményeim szerint dolgozatom hasznos képet fog nyújtani olvasója számára az aktív portfoliókezeléshez használható adatbányászati módszrekről, azok előnyeiről, hátrányairól és alkalmazási lehetőségeiről.

1. Tőzsdei idősorok jellemzői, előrejelzési módszerei

A tőzsdei idősorok alakulása már régóta az egyik legfontosabb információ a befektetők számára, azonban az elmúlt években még nagyobb jelentőssége lett a

pénzügyi/befektetési döntések meghozalatalában köszönhetően a világszerte jellemző alacsony kamatkörnyezetnek. Ennek köszönhetően a tőzsdei idősorok előrejelzésének lehetőségének vizsgálata a magán- illetve intézményi befektetők és spekulánsok mellett az elmúlt évtizedben a kutatók: matematikusok, fizikusok, informatikusok, közgazdászok figyelmét is felkeltette. A kutatók számára a probléma megoldást nagyban nehezíti hogy a pénzügyi idősorokra jellemző hogy zajosak nemstacionáriusak, nemlineárisak és kaotikusak, és gyakran fordul elő bennük strukturális törés, így nagy kíhívás pontosan előrejelzni őket (Hall, 1994; Li, et al., 2003; Yaser & Atiya, 1996, Huang et al., 2010; Lu et al., 2009, Oh & Kim, 2002; Wang, 2003). Ezenkívül a tőzsdei idősorok alakulását nagyban befolyásolják makroökonómai faktorok változásai, politikai események, a befektetők elvárásai, más pénzügyi termékek áralakulása, banki kamatok változása és a befektetőki pszichológa is(Tan et al., 2007). Ezen okok miatt a pénzügyi/tődzsdei idősorok előrejelzése az egyik legnagyobb kihívás a pénzügyi piaci szereplők számára.

A tőzsdei adatok előrejelzésével kapcsolatban kétfajta hipotézis létezik. Az első hogy a piacok minden pillanatban hatékonyak (Efficient Market Hypothesis EHM), azaz a részvények áraiban az összes ismert információ megjelenik. Emiatt az árak változása véletlenszerű, mivel csak az új információk hatnak rá, így a részvények árfolyama véletlen bolyongási (Markov) folyamatot követnek (ugyanolyan valószínűséggel nőnek, mint csökkennek). Ha elfogadjuk ezt a feltételezést akkor az állítjuk hogy az árfolyamok alakulásában nem figyelhető meg trend vagy bármilyen a korábbi időszakhoz hasomló minta és így a piaci szereplőknek nincs esélyük előrejelzni az ezek várható alakulását (Sitte & Sitte, 2002. Az elméletet rengeteg kritika érte az elmúlt években és egy másik hipotézis is kialakult nem hatékony piacok elmélete névvel (Inefficient Market Hypothesis IHM) (Pan, 2003). Ez azt állítja hogy a pénzüigy piacok legalább alkalmanként nem hatékonyak, a piac nem véletlenszerűen mozog így van esély bizonyos modellekkel előrejelezni az árfolyamok alakulását. A tény hogy bizonyos szereplők folyamatosan átlagon felül tudnak teljesíteni a piacon azt sejteti hogy a hatékony piacok elmélete a valóságban nem vagy nem teljesen állja meg a helyét (Fama, 1970). A hatékon piacok lézetését Grosmann & Stiglitz (1980) elméleti síkon is cáfolta, ugyanis azt állították hogy mivel tökéletesen hatékony piacon információszerzéssel nem lehetne átlagon felüli profitra szert tenni, ezértnem is fog senki erre energiát fordítani, így a beérkező információk sem rögtön épülnek be az árba (Grossman-Stiglitz paradoxon).

Az utóbbi elmélet miatt elmúlt két-három évtizedben egyre növekvő számú tanulmány jelent meg az akadémai szférában amelyben a pénzügyi instumentumok mozgázást próbálják előrejelezni. Ezeket a tanulmányokat a befektetők és spekulánsok tudják nagyon könnyen tudják hasznosítani, és így profitálni az egyre szélesedő akadémiai kutatásokból, hiszen ha elég pontosan tudják megállapítani hogy melyik részvény hogyan változik a következő időperidósuban akkor megfelelő befektetési stratégiák alkotásával átlag feletti hozamot tudnak elérni a piacon (Chen et al., 2003).

Az előrejelző módszereket amiket ezekben a tanulmányokban használnak két kategóriába lehet osztani: statisztikai/ökonometrai és adatbányászati/gépi tanulási módszerek. A tradicionális statisztikai módsezrek közé tartozik a lineáris regresszió a mozgó átlag, az exponenciális simítás, az ARIMA, a GARCH és VAR. Ezek a módszerek akkor adnak jó előrejelzési eredményeket, ha a pénzügyi idősorok lineárisak vagy közel lineárisak, azonban a való életben ahogy korábban is említettem a tőzsdei idősorokra a nemlinearitás a jellemző. A hiba oka, hogy ezek a módszerek azon a feltevésen alapulnak, hogy a pénzügyi idősorok között lineáris korrelációs struktúra van, így nemlineáris mintázatokat nem tudják felismerni és így előrejelzni sem (Khashei et al., 2009).Emellett a hagyományos statisztikai módszerek nagymennyiségű historikus adatot követelnek, és a jó előrejelzési eredményhez emellett megkövetelik azt is hogy ezek eloszlása normális legyen (Cheng & Wei, 2014).

Ezeket a feltételezést küszöbölik ki az adatbányászati módszerek amelyek jobban tudják modellezni az idősorok nemlineáris struktúráját. Ide soroljuk a neurális hálók mellett a tartóvektor gépeket (SVM) ésa döntési fák különböző fajtáit is. Ezek adatvezérelt és nem-parametrikus módszerek tudnak ismeretlen kapcsolatokat feltárni és kezelni az empirikus adatok között, így hatékonyabban tudják előrejelzeni a bonyolult és nemlineáris tőzsdei adatok változását. (Chen et al., 2003; Chun &Kim, 2000; Thawornwong & Enke, 2004;Enke & Thawornwong, 2005; Hansen & Nelson 2002) Az elmút években megjelenő egyre több adatbányászati cikk és alkalmazás is azt mutatja hogy ezek a módszerek versenyképesek és jelentős előnyeik vannak a hagyományos mdószerekhez képest (Lu et al., 2009; Duan & Stanley, 2011; Huang et al. 2010 ;Ni & Yin 2009). Azonban ezen modelleknek is megvannak a hátrányai, többek között a neurális hálók alkalmazásakor gyakori probléma a túltanulás és hogy a lokális minimumot találja meg a globális helyett, a tartóvektor gépek és döntési fák pedig érzékenyek a paraméterek megválasztására, ezenkívül ezek optimális beállítása miatt

az előrejelzés szükséges ideje jelentősen hosszabb, mint az ökonometriai modelleknél. (Wang et al., 2012, Yu et al., 2008)

Az idősorok nem stacioner tulajdonsága miatt a függő váltózó és a magyarázó változók kapcsolata az idő múlásával megváltozhat, azonban a legtöbb adatbányászati algoritmus alapja hogy a változók között konstans kapcsolat van. (Cao & Gu, 2002) A struktúrális változások, amelyeket általában politikai események vagy a befektetők várakozásainak megváltozásai okoznak, jellemzőek a pénzügyi paicokra, így nagy kihívás ezek megfelelő figyelembe véltele a modellezés során. Ennek egyik megoldása az egyes adatbányászati technikák keresztezése (hybridálása) ami az elmúlt években egyre jellemzőbb az akadémai körökben. Az alapötlet, hogy a hybrid módszerek kiküszöbölik az egyedi módszerek hátrányait és szinergiát alkotva javítják az előrejelzések pontosságát (Lean et al., 2008).

A módszernek alapvetően három különböző fajtája van. Az első a „divide-and conquer" elven alapszik, aminek lényege hogy komplex problémák esetén érdemes lehet több kisebb problémra felosztani azt, majd megoldani őket. Ennek egyik legelterjedtebb alkalmazása a tőzsdei előrejelzésben az EMD (empirical-mode-decomposition), amikor az idősort oszcilláló idősorok öszegére bontjuk szét (IMF-ek), majd ezeket külön-külön előrejelzzük, és utána aggregáljuk, hogy megkapjuk az eredeti idősor előrejelzett értékét. (Cheng & Wei, 2014) A második esetben megpróbáljuk kiszűrni a modellek input változóiból a zaj-t, ezáltal elősegítve hogy pontosabb eredményt kapjunk. Erre a leggyakrabban a főkomponens-analízist (PCA) és a független komponens analízist (ICA) használják, amelyek lényege hogy az input változókból független komponenseket létehozva (IC-k) megállíptható hogy melyik komponens tartalmazza a zajt és az eltávolítva növelni tudjuk az előrejelzés pontosságát. (Lu et al., 2009) A harmadik módszer pedig a különböző adatbányászati modellek előrejelzéseinek kombinálása az egyszerű aggregálástól kezdve a bayes-i átlagoláson át a Lasso regresszióig. A kombinálási módszerek alapja, hogy az egyes módszerek kombinálásával az előrejelzés varianciája csökkenthető. (Sermpinis et al., 2012).

2. Az adatbányászat és annak módszertana

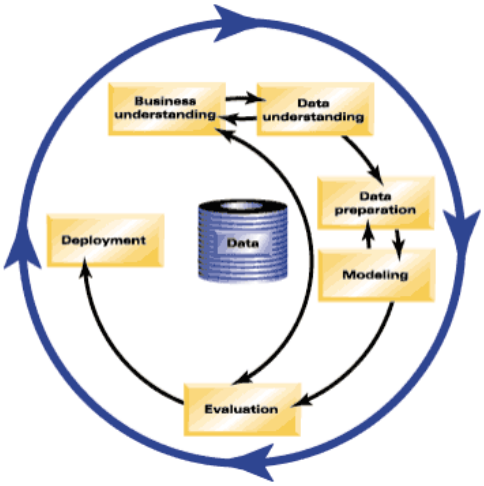
Az adatbányászat az informatikai szektor egyik leggyorsabban fejlődő iparága. Mint tudományág az üzleti intelligencia (BI-Business Intelligence) része és célja hogy olyan rejtett

összefüggéseket derítsen ki az adatokból, amelyek üzletileg hasznosíthatóak. A mérnöki gyakorlattól az üzleti életen át a tudományig szinte mindenhol használják, ahol nagy mennyiségű adatot kell elemezni és azok alapján modelleket alkotni. A pénzügyi területen belül hitelbesoroló modelleknél, csődelőrejelzésnél és pénzügyi idősorok vizsgálatánál is használjuk.

Az adatbányszati projektek kulcspontja amegfelelő módszertan megválasztása. A leggyakrabban használt módszertan a CRISP-DM (Cross Industry Standard Process for Data Mining) amely 6 fázisból tevődik össze (IBM, 2011).

1. Üzleti folyamatok megértése (Business Understanding)
2. Adatok megértése (Data Understanding)
3. Adatok előkészítése (Data Preparation)
4. Modellezés (Modelling)
5. Ellenőrzés (Evaluation)
6. Alkalmazás (Deployment)

Ezek azonban nem lineáris követik egymást, hanem visszacsatolások figyelhetőek meg az egyes folyamatok között amelyet az 1. ábra mutat be.



1.ábra Az adatbányászati módszertan felépítése (IBM, 2011)

Az adatbányászti projetek első feladata, hogy megértsük a üzleti problémát, meghatározzuk az célokat és a sikerkritériumokat. A következő lépésben megvizsgáljuk milyen adatokat tudunk beszerezni az adott kérdés vizsgálatához és hogy azok milyen minőségnem érhetőek el. Ezután következik az adatok előkészítése a modellekhez. Erre azért van szükség, mert különböző modelleknek más-más fajta input változókra van szükségük

(SVM esetén nem célszerű túl sok változót bevonni mert az rontja a módszer hatékonyságát, míg a döntési fák kevésbé érzékenyek erre), és ha az adatokat nem készítjük elő megfelelően a modellezéshez akkor a kapott előrejelzések elmaradhatnak a célként kitűzött eredménytől. Általában az adatbányászati projektek esetén az idő legalább 60%-át az adatok előkészítésére tisztítására kell fordítani. A következő lépsében meghatározzuk, hogy milyen adatbányászati módszereket alkalmazunk és hogy ezeket milyen paraméterbeállítások mellett használjuk. A kapott eredményekről el kell dönteni hogy használhatóak-e a kitűzött cél eléréséhe és ha igen akkor az készített modelleket valós környezetben is elkezdhetjük használni. (Petróczi, 2009)

A tőzsdei idősorok előrejelzése több szempontból is eltér a standard adatbányászáti projektektől. Egyrészt az adatok előkészítéséhez lényegesen kevesebb idő szükséges, hiszen szinte minden részvényre jó minőségű (nem nagyon van hiányos adat közöttük) magas frekvenciájú adat tölthető le több helyről is, másrészt viszont a modellezés során a legjobb algoritmusok kiválsztása hosszadalmas lehet. Utóbbi oka, hogy míg egy standard adatbányászati projekt esetén egy adott szint felett nem érdemes már javítani a módszereket (viszonylag kis nyereség érhető el velük sok többletenegia befektetésével), addig itt egy-egy tizedszázaléknyi javulás is jelentős profitot termelhet a későbbiekben. Emiatt érdmes a lehető legtöbb adatbányászati módszert kipróbálni és azok optimális paraméterbeállításait megkeresni.

Tőzsdei idősorok előrejelzése esetén a megfelelő módszerek algoritmusok kiválasztása során többféle kihívással kell szembenéznie a modellezőnek (Moody, 1995):

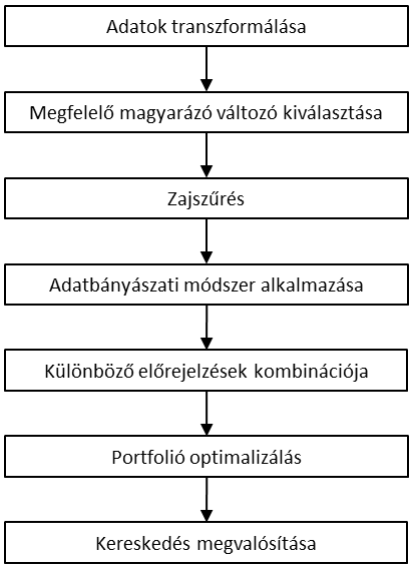
- **Változók kiválasztása:** A mai napig nincs egyértelmű szabály és konszenzus arra nézve sem az akadémai körökben sem a gyakorlati szakemberek között, hogy milyen változókat érdemes használni a modellezés során, és hogy milyen összefüggések alapján lehet ezeket kiválasztani. A jó predikciós eredményt adó változók megtalálása a legelső és egyik legnehezebb feladat.
- **Zaj kiszűrése:** Ahogy az előző fejezetben említettem a pénzügyi/tőzsdei idősorok zajosak, így ha ezt nem kezeljük a modellezés során, akkor az előrejelzés erdményessége jeletősen csökkenhet. Nehezíti a problémát hogy magasabb frekvenciájú adatok esetén a jel-zaj-arány egyre rosszabb (Kondratenko & Kuperin, 2003).

- **Dinamikus komplex rendszer modellezése:** A pénzügyi idősorokra jellemző hogy egy komplex dinamikusan változó rendszert alkotnak, amely jelentősen megnehezíti azok előrejelzését.
- **Nemlineráis adatok modellezése:** Ahogy korábban említettem az idősorok jellemzően nemlineárisak így a hagyományos statisztikai módszerek alkalmazása helyett megfelelő adatbányászati modelleket érdemes használni. Azonban ezek között is vannak kevésbé és jobban alkalmazható modellek, amelyeket figyelembe kell vennie a modellezőnek az előrejelzés során (Nebehaj, 2010).

Egy tőzsdei előrejelzésen alapuló aktív portfoliókezelő stratégia megalkotása során tehát a következő kihívásokkal nehézségekkel kell szembenézni:

1. megfelelő magyarázó változók kiválasztása (***feature selection***)
2. zajszűrés, jelfeldolgozás (***financial signal processing***)
3. valamilyen adatbányászati módszer alapján előrejelzés a paraméterek optimalizálása mellett (***forecasting with data mining methods***)
4. különböző előrejelzések kombinálása (***combining data mining techniques***)
5. az előrejelzett részvényárfolyamok alapján az optimális portfolió megalkotása (***portfolio optimalization***).

A teljes folyamatot beleértve az adatok előkészítését, transzformálását és a kereskedést a 2. ábra mutatja.



2.ábra A dolgozatban bemutatott modellkeret felépítése (Saját szerkesztés)

A következő fejezetben bemutatom hogy ezen lépések során milyen módszereket lehet és érdemes felhasználni.

3. Az adatbányászati modell felépítésének folyamata

3.1. Lehetséges inputváltozók kiválasztása

Az előrejelzéshez szükséges inputváltozók számáról nincs egyhangú/egyöntetű konszenzus, vannak olyan esetek, amikor csak két változót használtak, de vannak olyanok is, amikor több mint ötvenet.(Constantinou et al., 2006; Ollson & Mossman, 2003) A leggyakrabban használt inputváltozók a részvény nyitó és előző időszaki záró ára, valamint a napi legalacsonyabb és legmagasabb értéke. A kutatások nagyjából 30%-ában szerepel a részvény korábbi időszaki záró ára vagy valami technikai indikátor amely felhasználja ezt az információt (Barnes at al., 2000; Halliday, 2004). Ezek mellett gyakori még korábbi napok tranzakciós volumeneinek használta is. Sokszor az előbb említett változókat piaci indexek (Dow Jones vagy S&P) vagy devizaárfolyamok adataival kombinálják (Huang et al., 2005; Phua et al., 2001). Feltörekvő piaci részvények/indexek előrejelzése esetén is jellemző hogy a részvény korábbi adati mellett bevonják a főbb devizák illetve tőzsdék árfolyamváltozásait is (Wikowska, 1995). A kutatások 20%-ában technikai indikátorokat használnak, aminek száma általában 2 és 25 között mozog (Armano et al, 2004). Gyakran fordul elő az is hogy a korábban említett változócsoportok közül mindegyiket alkalmazzák a modellezők, ennek egyik extrém esete Kosaka et al. (1991) cikke, amelyben 300 magyarázó változót használnak az előrejelzéshez (Atsalakis &Valavanis, 2009).

3.2. Megfelelő magyarázó változók kiválasztása

A legtöbb esetben az inputváltozók kis száma is elég információval rendelkezik a pontos előrejelzéshez így elég csak ezeket használni. Természetesen jogos feltételezés lehet hogy több változó több információval szolgál, így javítja az előrejelzés pontosságát, ha mindegyiket használjuk, azonban ha van olyan változó, ami irreleváns, akkor annak bevonása ronthatja az előrejelzés pontosságát. A részvényárak előrejelzésekor a nagy dimenziójú inputváltozók egyrészt növelik a számításigényt, másrészt annak az esélyét is hogy a modell túltanul. A inputváltozók szűkítése esetén tehát dimenziócsökkentési probléma keretein

belül keressük meg azokat a változókat, amelyek a legjobban használhatóak az előrejelzés során (Liu & Zheng, 2006) A modellbe bevonandó magyarázó változók kiválasztásakor arra törekszünk, hogy minél kevesebb inputtal tudjuk minimalizálni az előrejelzés hibáját (Koller & Sahami, 1996).

Vannak olyan adatbányászati modellek amelyek érzékenyek az input változók számára (SVM), és vannak amelyek előrejelzési pontossága független ettől (döntési fák). A megfelelő magyarázó változók kiválasztására az egyik leggyakrabban használt módszer a stepwise regression alapján való döntés (stepwise regression analysis Esfahanipour & Ahamiri; 2010; Hadavandi & Shadavandi, 2010), de emellett használnak korreláció-alapú változó-kiválasztást (correlation-based feature selection Huang & Tsai, 2009) és informciónyereség-alapú eljárást is (information-gain Ni et al., 2011). Ezek mellett az egyik leghatékonyabb eszköz a dimenziószám csökkentésére a főkomponens-analízis (PCA). Ekkor az eredeti változókat lineáris transzformáció segítségével új korrelálatlan változókba transzformáljuk, majd az utolsók közül néhányat elhagyunk mivel sorrendben a teljes variancia egyre kisebb hányadát magyarázzák a kapott komponensek. A módszer nagy előnye, hogy nemcsak a magyarázó változók számát csökkentjük, hanem egyúttal az azokban lévő zajt is. (Petróczi, 2009; Dai et al., 2012)

3.3. Adatok transzformálása

A legtöbb input változó adatai széles skálán mozognak így az adatok transzformálása szükséges, hogy ne csökkentsük a tanítási módszerek hatékonyságát. Több tanulmány köztük Atsalakis & Valavanis (2009) is rámutat, hogy az adatok elő-feldolgozása javítani tudja a modellek előrejelzési képességét, például neurális hálók esetén gyorsítja a tanulás konvergenciáját. Ezeket a transzformációkat adat-normalizálásnak is nevezzük, amelynek 3 fő változata van: az adatok logaritmizálása, illetve a változók skálázása a [0,1] vagy a [-1,1] intervallumba, vagy azok standardizálása.

3.4. Alkalmazott mintahossz kiválasztása

A megfelelő mintanagyság kiválasztása az egyik legfontosabb kulcspontja az előrejelzésnek. Vannak olyan szerzők akik fontosnak tartják a nagy elemszámú mintát mert

csak így áll elegendő információ a pontos előrejelzéshez (7000, Kanas & Yannopoulos, 2001), viszont vannak olyanok is akik ezt kevésbé tartják fontosnak és a modellezés meggyorsítása érdekében kisebb elemszámú mintával dolgoznak (40, Chaturvedi and Chandra, 2004). Előbbi hátránya hogy ennyi idő alatt már strukturális változások is lehetnek a pénzügyi piacokban, utóbbinak pedig hogy túl kicsi minta bizonyos extrém eseteket (tőzsdei zuhanások) nem tud előre jelezni (Varga, 2009). Ezen okok miatt a legtöbb kutatás 2000-2500 megfigyelést használ fel a modellezéshez.

3.5. Tanuló, tesztelő, validáló halmaz méretének kiválasztása

A legtöbb adatbányászati módszer tanuló algoritmus segítségével használja fel az árfolyam múltbeli mintáit az előrejelzéshez, ezért a túltanulás elkerülése miatt az előrejelzés során az adathalmazt 3 részre kell osztani. Az első a tanuló halmaz amely általában a minta 60-70%-a, és a modellek az erre jellemző mintákat a tanulási algorimusok segítségével tudják általánosítani és felhasználni az előrejelzés során. A validációs halmaz (a minta 10-20%-a) alapján értékeljük ki a hálózatokat, és ezen a halmazon mutatott teljesítmény alapján lehet kiválasztani az optimálisat. A teszthalmaz (az adatok 10-20%) szolgál az optimáls paraméterbeállítás mellett futtatott modellek végső kiértékelésére. Fontos megjegyezni egyrészt hogy a modellek teljesítményének kiértékelése során olyan mintát használjunk (out-of-sample) amelyet nem használtunk fel a tanítás során (in-sample), másrészt hogy a szakirodalomban a validáló és tesztelő halmazok fogalmát gyakran felcserélve használják (Nebehaj, 2010).

3.6. Zajszűrés és hybrid módszerek

Ahogy korábban említettem a részvényárfolyamok változását számos faktor, többek között a szezonális változások és gazdasági, gazdaságpolitikai, illetve politikai események befolyásolják. Mok et al., 2004 tanulmánya kimutatta, hogy a hozamok változására különböző gazdasági indikátorok, vállalati hírek pszichológiai faktorok mellett a politikai intézkedések is hatnak. Emiatt, hogy pontos előrejelzést tudjunk készíteni szükséges hogy a részvények árfolyammozgása mögötti látens változókat megtaláljuk és felhasználjuk a modellezés során. Az ilyen problémák megoldására a mérnöki gyakorlatban már elterjedt

módszer a független komponens elemzés (ICA) alkalmazható, amely képes arra, hogy feltárja az adatsorok változását befolyásoló rejtett komponenseket és ezeket különválassza egymástól, méghozzá úgy, hogy ezek a lehetős legkevésbé függjenek egymástól és lineáris kombinációjukból felírható legyenek az eredeti adatsorok (Kapelner & Madarász, 2012).

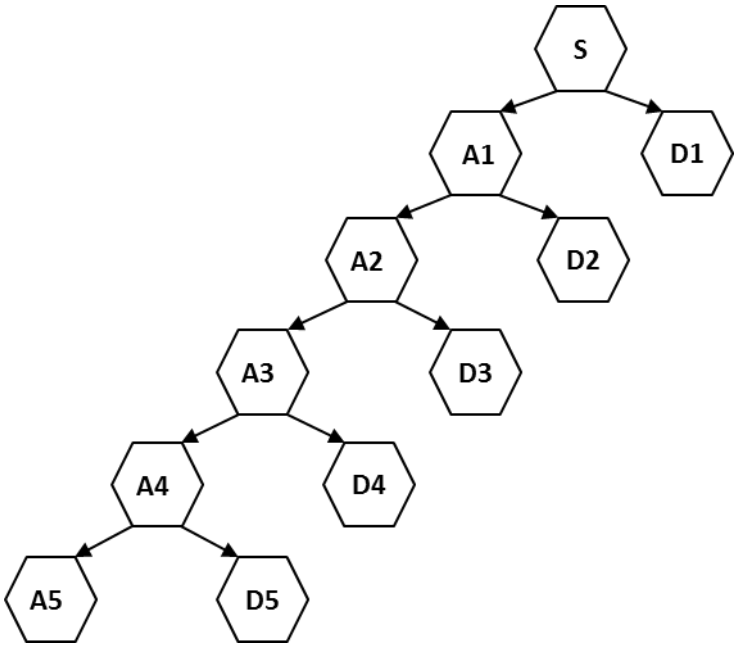
A független-komponens-analízis tehát egy olyan változó-transzformálós technika, amelynek célja, hogy változók alakulására ható a független faktorokat előállítsa a megfigyelhető „kevert" adatokból anélkül, hogy bármilyen előzetes ismeretünk lenne ezekről a faktorokról, és hogy tudnák a „keverés" struktúráját. A független komponensek (IC-k) a rejtett információi a megfigyelt adatoknak. (Hyvarinen et al., 2001) Ahogy említettem, a pénzügyi/tőzsdei idősorok zajosak és ez problémát okoz az előrejelzés során, mivel a zajt nem lehet közvetlenül kiszűrni a megfigyelt adatokból. Azonban a független komponens-elemzéssel lehetőség van arra hogy a megtaláljuk és eltávolítsuk a zaj komponenst a modellezéshez hasznát adatokból így javítva az előrejelzés pontosságát (C-J. Lu, 2010).

A módszer gyakran használják a műszaki életben jelfeldolgozásra (Beckmann & Smith, 2004), arcfelismerő rendszereknél (Déniz, Castrillón, & Hernández, 2003) zajszűrésre (James & Gibson, 2003) és természetesen tőzsdei idősorok előrejelzésére is. Oja et al., (2000) független-komponenselemzést használtak hogy csökkentsék a modell input-adatainak zaj/jel arányát majd autoregresszív modellt használva jelezték előre deviza-árfolyamokat. Cao & Chong (2002) pedig a zajszűrés után SVM modellt használtak pénzügyi idősorok előrejelzéséhez és megállapították, hogy a módszer használta jelentősen javítja az előrejelzést.

Pénzügyi adatok elemzésére a főkomponens-elemzést (PCA) is szokták használni, amelynek nagyon hasonló a módszertana a függetlenkomponens-analíziséhez, ezért érdemes összehasonlítani a két módszert. A PCA célja az adott adathalmaz főkomponenseinek megtalálása, amely a lehető legtöbb információt tartalmazza az adathalmaz összességéről. Ezek sorrendje fontossági sorrend is, így ha az utolsót, mint zajt elhagyjuk akkor kapjuk a legkisebb információveszteséget. Ezzel szemben az ICA a korrelálatlanság mellett a függetlenséget is feltételként szabja az egyes komponensek számára tehát a PCA egyfajta kiterjesztésének is tekinthető. Az ICA még abban is különbözik, hogy nem állít fel sorrendet az egyes komponensek között, számításukkor azok sorrendje felcserélődhet, így egy külön algoritmus szükséges hogy megtudjuk határozni melyik komponenst tekinthetjük zajnak (TnA eljárás) (Kapelner & Madarász, 2012). Több cikk is

összehasonlította a két módszert pénzügyi előrejelzés céljából és mindegyikben az állapították meg hogy az ICA használata jobban javítja az előrejelzést, mint a PCA (Back & Weigend, 1997; C-J. Lu et al., 2009; C-J. Lu, 2010), így dolgozatomban is ezt a módszert fogom használni zajszűrésre.

Egy kicsit más szempontból közelíti meg a zajszűrést az EMD dekompozíciós eljárás és a wavelet transzformáció amelyek nem az input változókból próbálják kiszűrni a zaj, hanem magából az eredeti idősorból. A wavelet transzformáció (a Fourier transzformció egy speciális típusa) esetén az eredeti idősort több iteráció alkalmával szétbontjuk alacsony és magas frekvenciájú jelekre. Miden egyes iteráció alkalmával a kapott alacsony frekvenciájú (mother wavelet) jelet bontjuk tovább a két komponensre. Ezt mutatja be a 3. ábra.



3.ábra A wavelet dekompozíció folyamata (Saját szerkesztés Wang et al., 2011 alapján)

Az alacsony frekvenciájú jel tartalmazza az idősor fő tulajdonságait, míg a magas frekvenciájút lehet zajnak tekinteni. Amikor egy adatbányászti modellel alkalmazzuk az előrejelzésre akkor a k-adik iteráció után kapott alacsony frekvenciájú jelet használjuk fel inputként. Ezen idősor korábbi értékei lesznek az adatbányászati modell inputjai (J.Z. Wang at al., 2011). A módszert a közgazdaságtan területén belül gyakran használják kőolajára előrejelzésére mind rövid (He et al., 2009, Silva et al., 2010, Tsung et al., 2011), mind hosszútávon (Yousefi et al., 2005).

A korábban említett „divide and conquer” elv az empirikus dekompozíció (EMD) lényege, amelyet Huang et al (1998) fejlesztett ki és a Hilbert-Huang transzformáción

alapszik. Ez az eredeti idősort véges számú IMF-ekre bontja fel amelyek könnyebben kezelhető frekvenciájúak és erősen korreláltak, így könnyebb egyesével előrejelezni őket, majd ezeket aggregálva megkapni az eredeti idősor előrejelzését(Cheng & Wei, 2014). Ezt a módszert gyakran használják földrengés-jelek dekompozíciójára (Vincent et al., 1999), szélsebesség (Guo et al., 2012) és akár turizmus előrejelzésére is (Chen et al., 2012). Az EMD hasonlít abban a waveletre hogy az eredeti idősort bontja szét komponensekre (IMF-ek) ugyanakkor ebben az esetben nem tudunk definiálni zaj komponenst és azt „kidobni" ezekközül, így ennek köszönhetően mindegyik komponenst használjuk az előrejelzés során.A módszernek a wavelet transzformációhoz képest több előnye is van: egyrészt egyszerű megérteni és implementálni, másrészt wavelet transzformáció esetén meg kell határoznunk már az elején hogy hányszor bontjuk fel az anya waveletet alacsony és magas frenkvenciájú jelekre és ennek nehézkes lehet a meghatározása bizonyos idősorok esetén. EMD alkalmazásakor ezzel szemben az algoritmus egyértelműen megmondja mikor kell abbahagynunk az iterációs lépéseket így ilyen problémával nem kell szembenéznünk (Yu et al., 2008). Dolgozatomban emiatt az ICA mellett ezt fogom kombinálni adatbányászati módszerekkel.

3.7. Lehetséges adatbányászati módszerek

A tradicionális statisztikai és ökonometriai módszerek korábban említett korlátai miatt az elmúlt két évtizedben rengeteg olyan tanulmány jelent meg, amelyben a kutatók pénzügyi/tőzsdei idősorokat jeleznek előre adatbányászati módszerekkel. Ezek közül az egyik legelterjedtebbnek és legnépszerűbbnek számítanak a különböző neurális hálózatok (Cao & Parry, 2009; Chang et al., 2009; Chavarnakul & Enke, 2008; Enke &Thawornwong, 2005), amelyek adatvezérelt, nem-parametrikus módszerek és nem követelnek erős modellfeltevéseket, valamint előzetes statisztikai feltételezéseket az input adatokról, továbbá bármilyen nem-lineáris függvényt tudnak modellezni (Vellido et al., 1999; Zhang et al., 1998). Az egyik legnagyobb előnyük, hogy nagyon komplex kapcsolatokat is megtudnak tanulni az input és output adatok között, még olyanokat is, amelyeket nehéz definiálni vagy akár ismeretlenek. Atsalakis & Valavanis (2009) közel 100 tanulmányt feldolgozó cikkében rámutat, hogy a különböző neurális hálózatok közül az előrecsatolt (feed forward neural network: FFNN) és rekurrens (recurrent neural networks RNN) hálókat alkalmazzák a

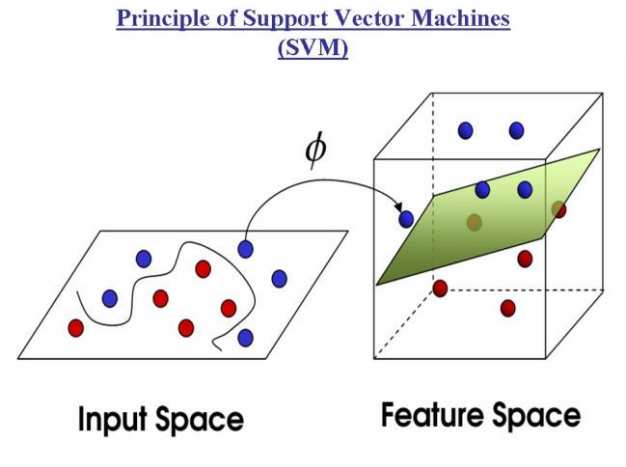
leggyakrabban a kutatók a pénzügyi idősorok előrejelzésére. Utóbbiak abban különböznek az előrecsatolt neurális hálóktól - amelyekben az inputok csak az előző rétegből érkeznek és csak a következő rétegbe továbbítódnak -, hogy memóriával rendelkeznek, azaz így nemcsak az input értékeket, hanem a saját időkésleltetett értékeit is használja a modellezés során (Nebehaj, 2010). Előrecsatolt neurális hálók közül a hiba-visszaterjesztéses (back-propagation neural network), míg rekurrens hálók közül az Elman és a Jordan halók a legnépszerűbbek.

Azonban ennek a gyakran alkalmazatott módszernek is vannak hátrányai, többek között az, hogy sok paraméterrel rendelkeznek, nehéz ezeket optimalizálni és nagy a kockázat a túltanulásnak és hogy a globális optimum helyett egy lokális optimumot talál meg a hálózat (Cao et al., 2001, Cao et al., 2003). Emellett a hálózatok csak olyan mintákat tudnak felismerni amely a tanuló halmazra jellemzőek, így ennek kiválasztása kulcsfontosságú, valamint egyes rekurrens hálókra jellemző, hogy lassú a konvergenciájuk, illetve magas a számítási igényük (Jaeger, 2002). Emellett legnagyobb hátrányként a modellek black-box jellegét szokták említeni, ami nagy korlátja az üzleti életben való alkalmazás elterjedésének, hiszen a modellek nem adnak pontos magyarázatot az előrejelzések okára, mivel a súlyok rendszere nem közvetlenül értelmezhető (Horváth, 2012).

Egy másik pénzügyi idősor előrejelzésre gyakran alkalmazott adatbányászati/gépi tanulási módszer a tartóvektorgépek (SVM), amelyet Vapnik (1995) fejlesztett ki és hamar népszerű lett a kitűnő általánosító képessége miatt, illetve hogy kis mintákon is eredményesen alkalmazható (Lee, 2009). A legnagyobb különbség a neurális hálózatok és az SVM között az a módszer, hogy előbbi az empirikus kockázatot minimalizálja (empirical risk minimalization ERM), míg utóbbi a strukturális kockázatot (structural risk minimalization SRM). Az SRM a felső korlátját minimalizálja az általános hibának ellenben az ERM-el ami a tanuló adatok hibáját. Így az SVM a globális optimumot találja meg és kiküszöböli a neurális hálók egyik korábban említett hibáját (Kim, 2003; Min & Lee, 2005). A módszer hátránya ugyanakkor, hogy nagyon érzékeny a modellbe bevont input változók számára.

Az SVM alapvetően egy osztályozó modell, amelynek feladata úgy fogalmazható meg, hogy két ponthalmazt válasszunk el egy olyan hipersíkkal, amelyhez maximális margó tartozik. Továbbá, ha feltesszük, hogy az adatok lineárisan szeparálhatóak, akkor is rengeteg megoldás létezik a lineáris elkülönítésre, így a módszer előnye, hogy olyan megoldást keres, amelynek a margó szélessége optimális. A lineárisan szeparálható feladatok csak egy részét

teszik ki az osztályozó feladatoknak, ezért egy kernelfüggvény bevezetésével akár magasabb dimenzióbeli, nemlineáris osztályozási feladat is megoldható, amely során a művelet számításigénye nem változik (Petróczi, 2009). Ekkor az adatokat egy magasabb dimenzióba transzformáljuk a kernel függvénnyel és megpróbáljuk megtalálni azt a hipersíkot, amely maximalizálja a margót az osztályok között (Hsu et al, 2009). A kernelfüggvény megválasztása kulcsfeladat a modellezés szempontjából leggyakrabban a radiális bázisfüggvényt használják a kutatók. A módszert lényegét a 4. ábra mutatja be.



4.ábra Az SVM adatbányászati módszer működése magasabb dimenzó esetén (Baban, 2008)

Azonban nemcsak osztályozási feladatokra alkalmazható a modell egy továbbfejlesztett változata az SVR (support vector regression), amely regressziós feladatokat tud modellezni (Vapnik et al., 1997). Utóbbit érdemes használni, amikor részvényárfolyamokat (Lu et al., 2009) vagy devizaárfolyamokat (Huang et al., 2010; Ni & Yin, 2009) modellezünk és figyelembe akarjuk venni a tranzakciós költségeket mivel egy 0-1 (fel-le) output esetén ezt nem tudnánk megtenni.

További megoldás lehet pénzügyi idősorok előrejelzésére döntési fák, genetikus algoritmusok használta, ezek irodalma még nem olyan széleskörű, mint az említett 2 modellnek. Előbbi esetén a rendelkezésre álló ismert változók eseteit halmazokra osztjuk, majd meghatározzuk hogy, ezek a képzett halmazok melyik végső csoporthoz tartoznak, így ezzel a módszerrel egyszerű döntések sorozatára vezethetjük vissza a bonyolult feladatokat. Előnye a módszernek, hogy képes kiválasztani az előrejelzés szempontjából fontos változókat, ugyanakkor hátránya, hogy jellemző rá a túltanulás (Petróczi, 2009). A döntési fáknak több fajtája létezik C5.0, C&RT, CHAID, QUEST és ID3 a különbség az közöttük, hogy hogyan definiálják a vágáshoz szükséges entropiát (Chang, 2011). Leggyakrabban ugyanakkor

egy nemrég kifejlesztett döntési fa algoritmust a Random Forestet alkalmazzák pénzügyi idősorok előrejelzésére (Qin et al., 2013). A genetikus algoritmusok az optimalizációs eljárások közé tartoznak és elsősorban a neurális hálók súlyainak (Asadi et al., 2012), illetve a különböző adatbányászati modellek paramétereinek (Kazem ez al., 2013) optimalizálására szokták használni ezeket/őket.

A módszerek nagy száma miatt nagyon időigényes lehet megtalálni, hogy egyes idősorok esetén, melyik a leghatékonyabb megoldás illetve ahogy láttuk mindegyiknek van előnye és hátránya is, ezért gyakran használnak többet a modellezés során, majd kombinálják ezek eredményeit. Én a dolgozatomban a neurális hálók közül az előrecsatolt változatot fogom használni a modellezés során.

3.8. Adatbányászati módszerek kombinálása

Hogy hogyan kombináljunk különböző előrejelzési technikákat az az idősor-előrejelzés irodalmának egyik legérdekesebb kérdése. Több kutató is rámutatott, hogy különböző technikákat, főleg rövid távú előrejelzés esetén érdemes kombinálni, és ennek előnye abból származik, hogy kiküszöböli az egyes módszerek hiányosságait (Zhang & Wu 2009, Armstrong 1989). Így akkor érdemes ezzel a módszerrel próbálkoznunk, ha nagyon eltérő adatbányászati/ökonometriai módszereket alkalmazunk az adatainkon. A módszerek kombinálása mellett érdemes a különböző tanuló algoritmusokat más-más halmazon tanítani. Habár Timmermann (2006) tanulmányában rámutatott, hogy egy egyszerű átlagolás is felveheti a versenyt a szofisztikáltabb technikákkal. Vannak olyan esetek, amikor az egyik módszer jóval pontosabb, mint a többi, így az átlagolás nem elég hatékony. Bates &Granger (1969) azt javasolta, hogy a kombinálási szabály az egyes előrejelzések varianciája-kovarianciája alapján történjen, míg Granger & Ramathan (1984) a regressziós technikát ajánlotta bíztató eredményekkel. Deutsch et al. (1994) lényegesen kisebb előrejelzési hibát ért el változó súlyú kombináció alkalmazásával, és ez további szofisztikáltabb módszerek kutatását indította el a 90-es években. A ridge regresszió (Chan et al., 1999) és a bayes-i átlagolás (Swanson & Zeng, 2001)mellett neurális hálókat és genetikus algoritmukat (Leigh et al., 2002) és Kalman-szűrőt (Sermpinis et al., 2012) is használtak a kutatók az előrejelzések pontosítása érdekében. Szinte minden szerző azt az állítást fogalmazta meg, hogy a különböző előrejelzési módszerek kombinálása szükséges, arról azonban nem született

egyezség, hogy mikor melyiket érdemes használni, így dolgozatomban többet is be fogok mutatni.

3.9. Portfoliókezelési módszerek

A portfoliókezlés az egyik központi problémája a pénzügyi irodalomnak és gyakorlatnak. A legismertebb portfoliókezelési elmélet az Markowitz (1952) átlag-variancia modellja amiben a portfoliót alkotó részvények kockázatát minimalizálja egy bizonyos vátható hozam mellett, vagy a probléma duálisát megfogalmazva maximalizálja a portfolió hozamát minimális kockázat mellett. A modell alapján a részvények közötti korrelációs mutató felhasználásával lehet eldönteni, hogy az egyes részvének hogyan járulnak hozzá a portfolió kockzatához. A modell feltételezi, hogy az összes részvény hozamának eloszlása normális és ennek átlag- és szórásparaméterét használja fel a részvények jövőbeli hozamának és kockázatának előrejelzésére. Annak ellenére, hogy a modell széles körben elterjedt, ezek az alapvető feltevések nem igazak a piaci adatokra, mivel a hozamok gyakran nemnormális eloszlásúak, így a variancia nem a legmegfelelőbb módszer a részvények kockázatának mérésére (Fama, 1965; Kon, 1984; Sharpe et al., 1999). Emellett az a feltételtezés, miszerint az előző időszaki átlagos hozam jó előrejelzés a jövőre nézve helytelenül veszi figyelembe a dinamikiáját a részvénypiacoknak, így pontatlan rövidtávú hozam- előrejelzéshez és a portfoliók alacsonyabb hozamához vezethet (Freitas et al., 2009). Így a portfolióoptimalizálásnak egy másik szemlélete is elterjedt, amely azon alapul, hogy a korábbi idősorok alapján empirikus úton próbálja meg előrejelzni az egyes részvények hozamait a egyszerű, számtani átlagnál szofisztikáltabb módszerekkel (Moody & Saffell, 2001; Pantazopoulos et al., 1998; Hellström, 2000). Ezeket nevezzük előrejelzés alapú portfolióoptimalizálásnak. Dolgozatomban mindkettőt bemutatom majd, így össze lehet hasonlítani a különböző portfoliók eredményességét az adott idősorokon.

4. A dolgozatban alkalmazott módszerek részletes bemutatása

4.1. Független komponenselemzés

Ha az adatbányászati módszereket úgy tanítjuk, hogy nem vesszük figyelembe hogy azok zajt tartalmazhatnak, akkor az ronthatja az általánosítás képességét a teszthalmazon,

illetve túltanuláshoz vezethet. Így az input adatok zajszűrése egy kiemelt feladat a modellezés során, amit én a független komponenselemzéssel fogok megoldani, és a következőekben ennek elméleti hátterét mutatom be.

Legyen $\mathbf{X}=[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m]^T$ egy többdimenziós adatmátrix mxn-es mérettel, ahol m≤n és a megfigyelt kevert jelek $\mathbf{x}_i$ mérete 1xn i=1, 2, ..., m. Az ICA modell alkalmazása esetén ez az **X** mátrix felírható a következő alakban

$$\mathbf{X} = \mathbf{AS} = \sum_{i=1}^{m} \mathrm{a_i}\, \mathrm{s_i}$$

ahol $\mathbf{a}_i$ az i-edik oszlopa a mxm méretű ismeretlen **A** keverőmátrixnak (mixing matrix) és $\mathbf{s}_i$ az i-edik sora az mxn méretű „source" **S** mátrixnak. Az $\mathbf{s}_i$ vektorok azok a látens adatok, amelyeket nem tudunk közvetlenül megfigyelni a kevert $\mathbf{x}_i$ adatokból, de utóbbiak ezen látens adatok lineáris kombinációjaként írhatóak fel (Dai et al., 2012). A független komponenselemzés célja, hogy megtaláljuk azt az mxm méretű W mátrix-ot (demixing matrix), amelyre teljesül, hogy

$$\mathbf{Y} = \mathbf{WX}$$

ahol $\mathbf{y}_i$az i-edik sora az **Y** mátrixnak i=1, 2, …, m és ezek a vektorok statisztikailag függetlenek (független komponensek). Ha a **W** mátrix az **A** keverőmátrix inverze $\mathbf{W}=\mathbf{A}^{-1}$, akkor a független komponenseket ($\mathbf{y}_i$) tudjuk használni, hogy megbecsüljük az eredeti látens jeleket $\mathbf{s}_i$ (C.-J. Lu, 2010).

Független komponenselemzés során egy optimalizációs problémát oldunk meg úgy, hogy megválasztjuk a független komponensek statisztikai függetlenségének egy objektív függvényét és optimalizációs eljárásokkal megkeressük a W mátrixot (C-J. Lu et al., 2009). Több ilyen kifejlesztett/kidolgozott eljárás létezik (Bell & Sejnowski, 1995; David & Sanchez, 2002; Hyvärinen et al., 2001), amelyek általában nem-felügyelt tanítási algoritmusokat használnak, hogy maximalizálják az IC-k statisztikai függetlenségét. A független komponensek nem normalitásából következik a statisztikai függetlenség, és ezt a nem-normalitást a következő normalizált differenciális entropiával (negentropiával) lehet mérni (Kapelner & Madarász, 2012):

$$\mathrm{J}(\mathbf{y}) = \mathrm{H}(\mathbf{y}_{\mathbf{gauss}}) - \mathrm{H}(\mathbf{y})$$

ahol $\mathbf{y}_{gauss}$ egy ortonomált vektor ugyanolyan kovariancia mátrixxal, mint **y**.

H egy olyan differenciális entropiája a véletlen **y** vektornak

$$H(\mathbf{y}) = -\int p(\mathbf{y}) \log p(\mathbf{y}) d\mathbf{y}$$

ahol p(**y**) az Y valószínűségi változó sűrűségfüggvénye. Ez a negentropia mindig nemnegatív és akkor és csak akkor nulla, ha **y** normális eloszlású. Mivel számítása nagyon nehézkes, ezért a következő egyenlet számításával lehet közelíteni:

$$J(\mathbf{y}) \approx [E\{G(y)\} - E\{G(v)\}]^2$$

ahol ***v*** egy normális változó nulla várható értékkel és egységnyi szórással, és ***y*** egy random változó hasonló várható értékkel és szórással. *G*-t általában a következő függvénnyel szokták megadni (C-J. Lu et al., 2009):

$$G(\mathbf{y}) = \exp(-\mathbf{y^2}/2)$$

Az ICA egyik leggyakoribb megoldási módja a Fast ICA algoritmus (Hyvarinen et al., 2001), amelyet én is alkalmazok majd a W mátrix definiálására.

4.2. Empirikus alapú dekompozíció (EMD)

Az empirikus alapú dekompozíció egy nemlineáris jel-transzformációs eljárás, amit (Huang et al., 1998) fejlesztett ki nemlineáris és nem-stacioner idősorok dekompozíciójára. Ez a módszer az eredeti idősort különböző időskálájú oszcilláló IMF komponensekre bontja fel (Yu et al., 2008). Ezek a komponensek alacsonyabb frekvenciájúak/kevésbé bonyolult képet festenek, így könnyebb előrejelzni őket. Minden egyes IMF-nek két feltételt kell kielégítenie: egyrészt lokális minimumok és maximumok összámának és a függvény null-helyeinek különbsége maximum egy lehet, másrészt a lokális átlagnak nullának kell lennie (Cheng & Wei, 2014). Ez az algoritmus a következő:

1, Határozzuk meg az összes lokális minimumát és maximumát **x(t)**-nek

2, Határozzuk meg az alsó $\mathbf{x_u(t)}$ és felső $\mathbf{x_l(t)}$burkolóját **x(t)**-nek

3, A felső és az alsó burkolót használva adjuk meg az idősor átlagát $\mathbf{m_1(t)=[x_u(t)+x_l(t)]/2}$

4, Számoljuk ki az eredeti idősor **x(t)** és az előző lépésben kapott átlag $\mathbf{m_1(t)}$ idősor különbségét $\mathbf{h_1(t)=x(t)-m_1(t)}$ami az első IMF-et ($\mathbf{h_1(t)}$) adja meg, ha kielégíti a fent említett két feltételt.

5, Miután megkaptuk az első IMF-et ugyanezt az iterációs algoritmust folytatjuk addig amíg nem kapjuk meg a végső idősort a reziduális komponenst **r(t),** ami egy monoton függvény és jelzi, hogy le kell állítanunk az algoritmust (Huang et al., 1999).

Ez eredeti idősort **x**(t) megkaphatjuk az IMF komponensek és a reziduális összegeként:

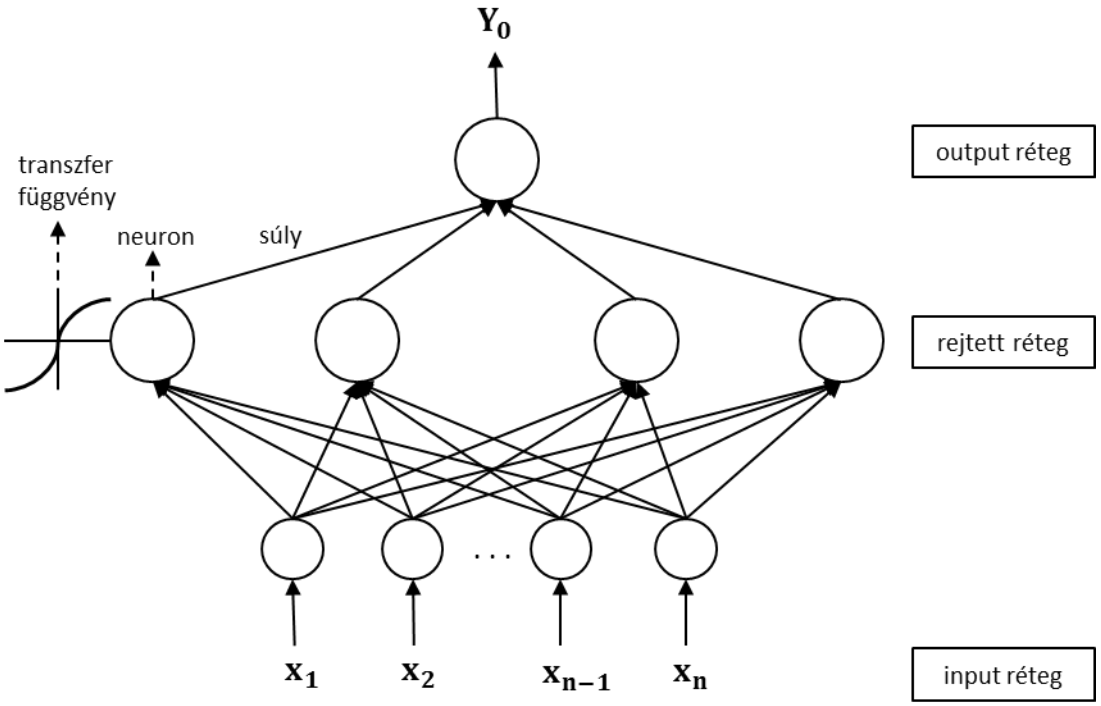
$$\mathbf{x}(t) = \sum_{i=1}^{n} h_i(t) + r(t)$$

A kapott IMF-ek közel ortogonálisak egymásra nulla közeli az átlaguk (Yu et al., 2008). A reziduális az eredeti idősor trend komponense, míg az IMF-ek csökkenő sorrendben egyre alacsonyabb frekvenciájúak (Cheng & Wei, 2014). Itt tehát az ICA vagy wavelet módszerhez képest nem mondhatjuk egyik komponensre sem azt, hogy zaj így nem is szükséges eltávolítani a komponensek közül.

4.3. Neurális hálózatok

A neurális hálózatoknak rengeteg formája létezik a pénzügyi idősorok előrejelzésénél. A leggyakrabban alkalmazott modell a többrétegű precepton hálózat (Multi-Layer Precepton MLP) (Kaasra & Boyd 1996) amely az előre csatolt neurális hálózatok közé tartozik. A hálózatok általában 3 vagy 4 réteggel rendelkeznek, az elsőt bemeneti rétegnek (input layer) az utolsót kimeneti rétegnek (output layer) míg a középsőket rejtett rétegeken (hidden layer) hívják. Minden egyes rétegben neuronokat találhatunk, az első réteg neuronjainak száma megfelel a modell magyarázó változói számának, míg az utolsó rétegé a célváltozó számával egyenlő (általában 2 neuron: bináris célváltozó, vagy 1 neuron: folytonos célváltozó). A rejtett rétegben lévő neuron száma határozza meg a modell komplexitását és az előrejelzési képességét. Ez előbb említett neuronokon kívül az inputrétegben és a rejtett rétegben még van 1–1 neuron (torzítás), amelynek értéke 1 és ugyanazt szerepet tölti be, mint a regressziós modell esetén a konstans. Normális esetben minden egyes neuron kapcsolódik az összes neuronhoz a következő rétegben és ezek az élek súlyokat reprezentálnak (Sermpinis et al., 2012). Az egyes neuronok az előző réteg neuronjaitól kapják az inputokat és azokat egy nem-lineáris függvényt használva alakítják át a következő réteg inputjaivá. Mivel egy rejtett réteggel rendelkező neurális háló bármilyen komplex problémát tud modellezni (Chauvin & Rumelhart, 1995), ezért én is ezt fogom használni a dolgozatom elemző/empirikus részében.

Egy háromrétegű hálózatot reprezentál a 5. ábra:



5.ábra Egy háromrétegű előrecsatolt neurális háló (Saját szerkesztés Dai et al., 2012 alapján)

A hálózat tanítása véletlenszerűen választott súlyok használatával kezdődik és a egy tanuló algoritmus (általában hiba-visszaterjesztő back-propagation) segítségével ezek a súlyok változnak az iterációk során. Az algoritmus célja, hogy megtalálja azokat a súlyokat, amelyek minimalizálják a hibafüggvényt (általában MSE RMSE vagy MAPE) a célváltozó és az aktuális változó között. A tanulás általában gradiens módszerrel történik, vagyis a súlyok értékét a hibafüggvény súlyvektor szerinti negatív gradiensének irányába módosítjuk (Varga, 2011) . Mivel a hálózat bizonyos számú neuronnal a rejtett rétegben bármilyen kapcsolatot meg tud tanulni a tanuló adatokon (akár az outliereket és a zajt is), ezért a tanuló algoritmus leállítási szabályával (early stopping) lehet megakadályozni hogy túltanuljon (overfitting) az adatokon. Emiatt ezt a tanuló eljárást a felügyelt tanuló eljárások közé lehet sorolni. Ezért szükséges adatok már a korábban említett tanuló tesztelő validáló halmazokra való felosztása. A tanítása a hálózatnak akkor áll le, ha a tesztelő halmazon a hiba eléri a minimumát. Ezután a teszthalmazon az adott paraméterű hálózatot kell lefuttatni (Sermpinis et al., 2012).

Legyen a hálózatunknak n input és m rejtett és egy output neuronja, ekkor a tanulási folyamatot a következő két lépcsőben lehet megadni (Zhang & Wu):

Első lépcső (rejtett réteg): A rejett réteg neuronjainak az outputját a következő egyenlettel lehet megadni:

$$\mathrm{net_j} = \sum_{i=0}^{n} v_{ij} x_i\,,\ \ j = 1,2,\dots,m, \text{ ahol}$$

$$y_j = f_H(\mathrm{net_j}), \qquad j = 1,2,\dots,m.$$

Itt a $\mathbf{net_j}$ az aktivációs értéke a j-edik neuronnak, $\mathbf{y_j}$ az outputja a rejtett rétegnek és $\mathbf{f_h}$ transzferfüggvény, ami általában sigmoid függvény szokott lenni:

$$f_H(x) = \frac{1}{1 + \exp(-x)}$$

Második lépcső: Az output réteg értékét a következő függvény adja meg:

$$O = f_O\left(\sum_{j=0}^{m} w_{jk} y_j\right)$$

ahol $\mathbf{f_o}$ a transzferfüggvény és ez leggyakrabban lineáris szokott lenni.

Mivel a tanítási eljárás során az első lépcsőben a súlyokat véletlenszerűen határoztuk meg, ezért többszöri futtatás esetén a hálózat különböző eredményeket adhat. Ezt a legtöbbször úgy szokták kiküszöbölni, hogy miden halózatot többször futtatnak le, majd az eredményeket átlagolják, így egyrészt robosztusabb eredményeket lehet kapni, másrészt az outlier hálózatokat ki lehet küszöbölni.

A neurális hálózatokban 2 paramétert: a rejtett rétegben lévő neuronok számát, illetve a tanulási tényező (learning rate) kell optimalizálni. Alacsony tanulási tényező esetén a tanulási folyamat lelassul a konvergencia előtt, míg magas tanulási tényező estén a hiba nem konvergál (Lu, 2010). Túl sok rejtett rétegben lévő neuron esetén a hálózat hajlamos a túltanulásra és a számítási igény is megnő, míg túl kevés esetén nem tudja a tanuló adatokon megfigyelt mintákat általánosítani. Az optimális paraméterek kiválasztására a keresztvalidációs (cross-validation) vagy a grid search optimalizációs eljárást szokták alkalmazni.

A szakirodalomban a neruális hálók tanítása során ahogy említettem a leggyakrabban a hibavisszaterjesztéses (BP) algoritmust használják, azonban bizonyos esetekben ennek lassú a konvergenciája, így újabb tanítási módszerek is megjelentek. A quick-propagation (QP) algoritmus mellett a Levenberg-Marquardt (LM) algoritmus is nagyon népszerű lett. Mindkettőnek nagy előnye, hogy a konvergenciája a tanulásnak sokkal gyorsabb, mint a BP algoritmusnak, mivel kevesebb iteráció szükséges hozzá (Asadi et al., 2012), amely kis hálózatok és rövid idősor esetén nem nagy előny, azonban jellemzően pénzügyi idősorok

esetén a minta elemszáma akár több ezres is lehet, így itt érdemes ezeket használni. Dolgozatomban az LM algoritmus fogom használni a hálózatok tanítása során.

4.4. Kombinációs technikák:

1, Egyszerű Átlagolás

Az első előrejelzés kombináló technika, amit alkalmazni fogok az azátlagolás, amit benchmarknak fogok tekinteni. A három előrejelző technikát adottnak véve f^t_{MLP}, $f^t_{ICA\text{-}MLP}$ és $f^t_{EMD\text{-}MLP}$ a t időpontban az átlag a következőképp alakul:

2, Bayes-i Átlagolás

Bayesi átlagolás esetén a kombinálás optimális súlyai az Akaike információs kritérium (AIC) vagy a Schwarz bayesi információs kritérium (SIC) alapján adódnak. A súlyokat AIC esetén a következő egyenlettel tudjuk számolni (Sermpinis et al., 2012):

$$w_{AIC,i} = \frac{e^{-0.5\Delta AIC_i}}{\sum_{j=1}^{3} e^{-0.5\Delta AIC_j}}$$

ahol i=1,2,3 az f^t_{MLP}, $f^t_{ICA\text{-}MLP}$ és $f^t_{EMD\text{-}MLP}$ reprezentálja, és

$$\Delta AIC_i = AIC_i - AIC_{i,min}$$

Ezek alapján a bayesi átlagoláson alapuló előrejelzés a következőképpen alakul:

$$f^t_{C_{NNs}} = \left(\sum_{i=1}^{3} w_{AIC,i} f^t_i\right)/3$$

Az Akaike információs kritérium az előrejelző modell relatív pontosságát adja meg és a következőképp számolható:

$$AIC = N\log\left(\frac{RSS}{n}\right) + 2k$$

ahol N az minta elemszáma k a paraméterek száma és RSS a rezidumok négyzetösszege (Panchal et al., 2010). A modell tehát a módszer pontossága mellett figyelembe veszi a komplexitást is a paraméterek számának kritériumba való beépítésével.

3, Granger és Ramanathan regressziós átlagolása (GRR):

Granger és Ramanathan 3 regressziós módszert javasolt a különböző előrejelzések kombinálására:

$$f_{c1} = a_0 + \sum_{i=1}^{n} a_i f_i + \varepsilon_1 \qquad [GRR-1]$$

$$f_{c2} = \sum_{i=1}^{n} a_i f_i + \varepsilon_2 \qquad [GRR-2]$$

$$f_{c3} = \sum_{i=1}^{n} a_i + \varepsilon_3, \qquad \text{ahol} \sum_{i=1}^{n} a_i = 1 \qquad [GRR-3]$$

ahol: f_i *i=1, 2, 3* az egyéni előrejelzések, azaz f^t_{MLP}, $f^t_{ICA\text{-}MLP}$és $f^t_{EMD\text{-}MLP,}$

f_{GRR1}, f_{GRR2}, f_{GRR3} a három regressziós előrejelzés,

α_0 a konstans a regresszióban,

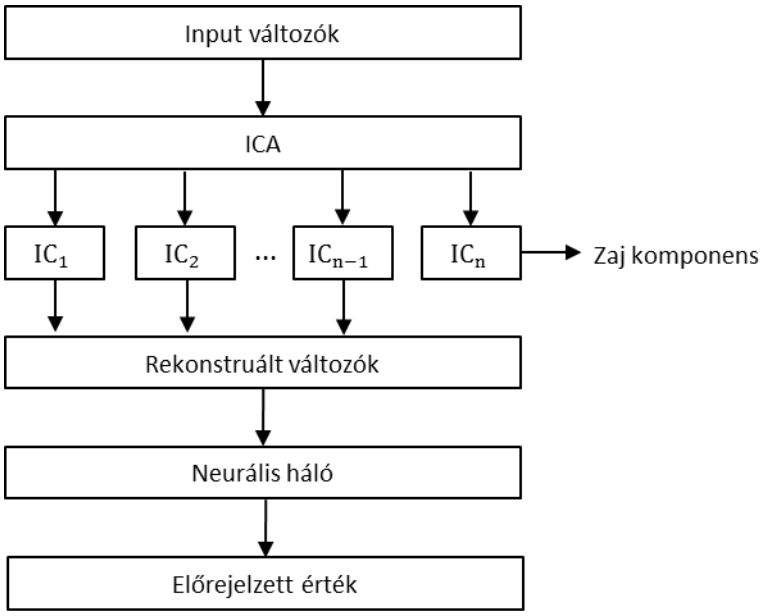
α_i a regressziós együtthatók az összes modellben,

ε_1, ε_2, ε_3, a hibatagok az egyes modellekben.

A legtöbb esetben a GRR-1 modellt használják az irodalomban (Swanson & Zeng, 2001), így én is azt fogom.

4.5. ICA-BPN modell

A következő hybrid modell, amit én is alkalmazni fogok dolgozatomban, három lépésből épül fel: először ICA módszer segítségével meghatározza az input változók független komponenseit (IC-ket), majd ezekből TnA módszerrel kiválasztja a zaj komponenst és ezt kiszűri, és végül BPN neurális háló segítségével előrejelzi az idősort. Ezt a folyamatot mutatja be a 6. ábra:



6.ábra Az ICA-BPN modell folyamatábrája (Saját szerkesztés)

Mivel a részvényárfolyamok esetén a zaj tartalmazza a legkevesebb információt a trendről, illetve az adatsor jellemzőiről, ezért azt a független komponenst tekintjük zajnak, amely a legkevésbé járul hozzá az adatstruktúra jellemzőihez (C.-J. Lu, 2010). Hogy megtaláljuk azt az IC-t amely a zajt reprezentálja, a TnA (testing-and-acceptance) módszerrel határozhatjuk meg, amelyet Cheung & Xu (2001) fejlesztett ki. Ez a módszer az RHD (relative hamming distance) hibát használja arra hogy definiálja a zaj komponenst. A módszer lényege, hogy minden lépésben 1–1 független komponenst kihagy és megvizsgálja, hogy anélkül visszaállított adatstruktúra mennyire tér el az eredetileg megfigyelt adatoktól. Az első iteráció esetén az utolsó független komponenst elhagyjuk és keverőmátrix segítségével próbáljuk rekonstruálni az eredeti megfigyelt mátrixot. Legyen $\mathbf{y_k}$ az utolsó függetlenkomponens, amit elhagyunk és a nélküle konstruált mátrix $\mathbf{X^R}$, amelyet a következő egyenlőség alapján kapunk meg:

$$X^R = \sum_{i=1, i\neq k}^{m} a_i\, y_i, \qquad 1 \leq k \leq m$$

ahol $\mathbf{X^R}=[\mathbf{x^R_1}, \mathbf{x^R_2}, \ldots, \mathbf{X^R_m}]^\top$ az mxn-es méretű rekonstruált mátrix, $\mathbf{x^R_i}$ az i-edik rekonstruált input változó $\mathbf{a_i}$ az i-edik oszlopa az **A** keverőmátrixnak, $\mathbf{A=W^{-1}}$, és $\mathbf{y_i}$ az i-edik független komponens (IC). Ezután az RHD hiba a rekonstruált $\mathbf{X^R}$ és az eredeti adatmátrix **X** között a következőképp számolható (Cheung & Xu, 2001):

$$RHD = \sum_{i=1}^{m} \left(\frac{1}{n-1} \sum_{t=1}^{n-1} [R_i(t) - R'_i(t)]^2 \right)$$

ahol:

$$R_i = sign[x_i(t+1) - x_i(t); \quad R'_i = sign[x_i^R(t+1) - x_i^R(t)$$

$$sign(r) = \begin{cases} 1 & \text{ha } r > 0 \\ 0 & \text{ha } r = 0 \\ -1 & \text{egyébként} \end{cases}$$

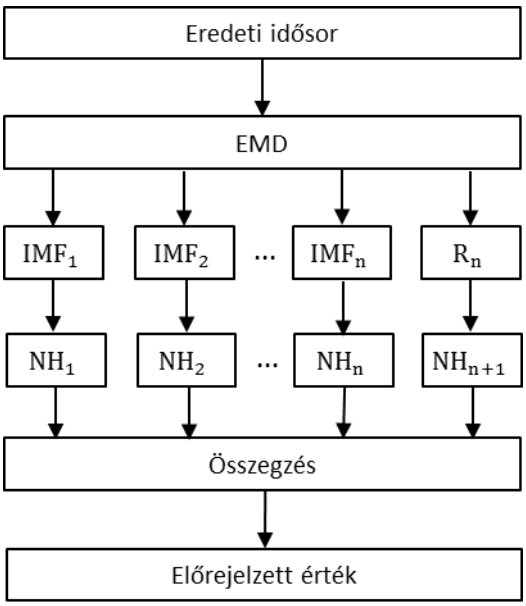
Az RHD hiba mértékét tehát arra használjuk, hogy megvizsgáljuk a hasonlóságot az eredeti idősor ($\mathbf{x_i}$) és a rekonstruált idősor ($\mathbf{x^R}$) között, azonos idősorok esetén ez a mutató nulla, teljesen különböző idősorok esetén pedig négy. Általánosabban megfogalmazva, ha ez a hiba közel van nullához, akkor a két idősor fő tulajdonságaiban azonosnak tekinthető, ha pedig távol van nullától akkor különbözőnek. Megvizsgáljuk, hogy melyik IC elhagyásával kapjuk a legkisebb értékű RHD hibát, és ezt tekintjük a későbbiekben zajnak, mivel ez a komponens adja a legkevesebb információt a rekonstruálás során. Ezt elhagyva az algoritmus megismételjük méghozzá m-1-szer és meghatározzuk a független komponensek sorrendjét. A legutolsó lesz a legfontosabb komponens, amely a legtöbb információt hordozza magában. Ezután a zajszűrt input változókat úgy kapjuk meg hogy a zaj komponens kivételével az összes független komponensre alkalmazzuk a korábban bemutatott rekonstruáló eljárást(C.-J. Lu, 2010).

Miután megkaptuk a zajszűrt előrejelző változókat, ezeket használjuk a BPN modell inputjaiként. Mivel ahogy írtam a korábbiakban egy rejtett réteg is elegendő hogy modellezzünk bármilyen komplex rendszert így két paramétere van a neurális hálózatnak amit optimalizálni kell: a rejtett rétegben lévő neuronok száma illetve a tanulási ráta. Ezek optimalizálására a grid search eljárást fogom alkalmazni dolgozatom során. Azt a paraméterpárt fogom alkalmazni a teszthalmazon amely minimalizálja az eltérések négyzetösszegének átlagát (MSE) a teszhalmazon.

4.6. EMD-BPN model

Az általam a dolgozatban használt másik hybrid modell is három lépésből épül fel: elsőként felbontjuk az eredeti idősort az EMD módszer szerint IMF komponensekre és a reziduálisra, ezután minden egyes IMF esetén egy BPN modell segítségével előrejelezzük a következő

időszaki értékeket, majd ezután az eredeti idősor előrejelzett értékét ezek összegeként konstuáljuk. Ezt a hybrid módszer mutatja be az 7. ábra:



7.ábra Az EMD-BPN modell folyamatábrája (Saját szerkesztés)

A módszer lépései:

1, Legyen **x(t)** az idősor, aminek értékeit előre akarjuk jelezni, és bontsuk ezt fel n IMF $\mathbf{h_i(t)}$, i=1,2,…,n és egy residuális komponens **r(t)** összegére a korábban bemutatott EMD dekompozíciós algoritmussal.

2, Az idősor dekompozíció után minden egyes IMF és a reziduális értékeit jelezzük előre egy előrecsatolt neurális hálózattal. Természetesen itt nem a másik két módszer által alkalmazott inputokat fogjuk használni (technikai indikátorok), hanem az egyes idősorok korábbi értékeit. Így ez az előrecsatolt neurális hálózat ekvivalens egy nemlineáris autoregresszív modellel (NAR modell) (Yu et al., 2005). Miden egyes idősor esetén meg kell határozni az optimális paramétereket (rejtett réteg neuronjainak száma, tanulási ráta, inputváltozók), amelyet az ICA-BPN modellhez hasonlóan grid search módszerrel fogok megkeresni. Mivel ebben az esetben az optimalizálás háromdimenziós (3 paramétere van a neurális hálóknak), és mivel gyakran 10–12 komponensre bomlik az eredeti idősor, ezért ennyiszer több neurális hálózatot kell optimalizálnunk, és így a számítási igény is jóval nagyobb lesz, mint egy adatbányászati modell esetén, azonban több kutatás is jelentős javulásról számolt be ezt a módszert használva, így mindenképp érdemes megvizsgálni (Yu et al., 2008).

3, Az egyes komponensek előrejelzett értékeit összeadjuk és így kapjuk meg az eredeti idősor előrejelzett értékét (Lin et al., 2012).

A bemutatott modell három fő része tehát a dekompozíció (EMD), előrejelzés (BPN), és az egyesítés (summerization). Több kutatási eljárásban az utolsó lépésben nem összeadják az egyes komponenseket, hanem átlagolják, vagy egy újabb neurális hálóval jelzik előre az eredeti idősor értékét (Yu et al., 2008).

4.7. Adatbányászaton alapuló aktív portfoliókezelés

Az aktív portfoliókezelést tekinthetjük kétlépcsősnek, ahol az első lépcső a részvények előrejelzése a második pedig az ezek alapján létrehozott befektetési stratégia megvalósítása. Ugyanakkor e kettő eredményessége erősen összekapcsolódik, hiszen rossz minőségű előrejelzésekkel nem, vagy csak nagyon nagy szerencse mellett tudunk átlag feletti hozamot elérni. Természetesen rengeteg befektetési stratégiát tudunk alkalmazni az előrejelzett részvényárfolymok segítségével, így ezek közül a legnagyobb profitot eredményezőt megtalálni komoly optimalizálási feladat.

Dolgozatoban négy stratégiát fogok alkalmazni és megvizsgálom hogy ezek közül melyik eredményezi a legnagyobb profitot hosszútávon két tőzsdei részvénnyel (OTP és MOL) való kereskedés esetén. Az első két stratégia esetén csak a részvények korábbi idősora alapján számolt szórást, várható értéket illetve korrelációt használom majd fel (Markowitz portfolió-optimalizálás), míg a másik három esetén egyre szofisztikáltabb módon az adatbányászati modellek által előrejelzett értékeket is. Dolgozatom során mindegyik stratégiát úgy alkotom meg, mintha a részvények helyett az azokra szóló egyszeri tőkeáttételes CFD-kkel kereskednék. Ezen derivatívák alkalmazásának előnye az, hogyha a részvény árfolyamának csökkenésére számítok, akkor nem kell eladnom a papírt, hanem megfelelő számú vételi kötést veszek a részvény eladását leképező CDF-jéből. A négy stratégia a következő:

1, A részvények várható értékét, szórását és korrelációját figyelembevéve a kereskedési időszak elején megállapítom a portfoliósúlyokat és ezen nem változtatok a kereskedési periódus végéig.
2, A kezdő tőkét fele-fele arányban szétosztom a részvények között, és ezután minden egyes nap végén az előrejelzett értékeknek megfelelően vagy long vagy short CFD-be fektetek.

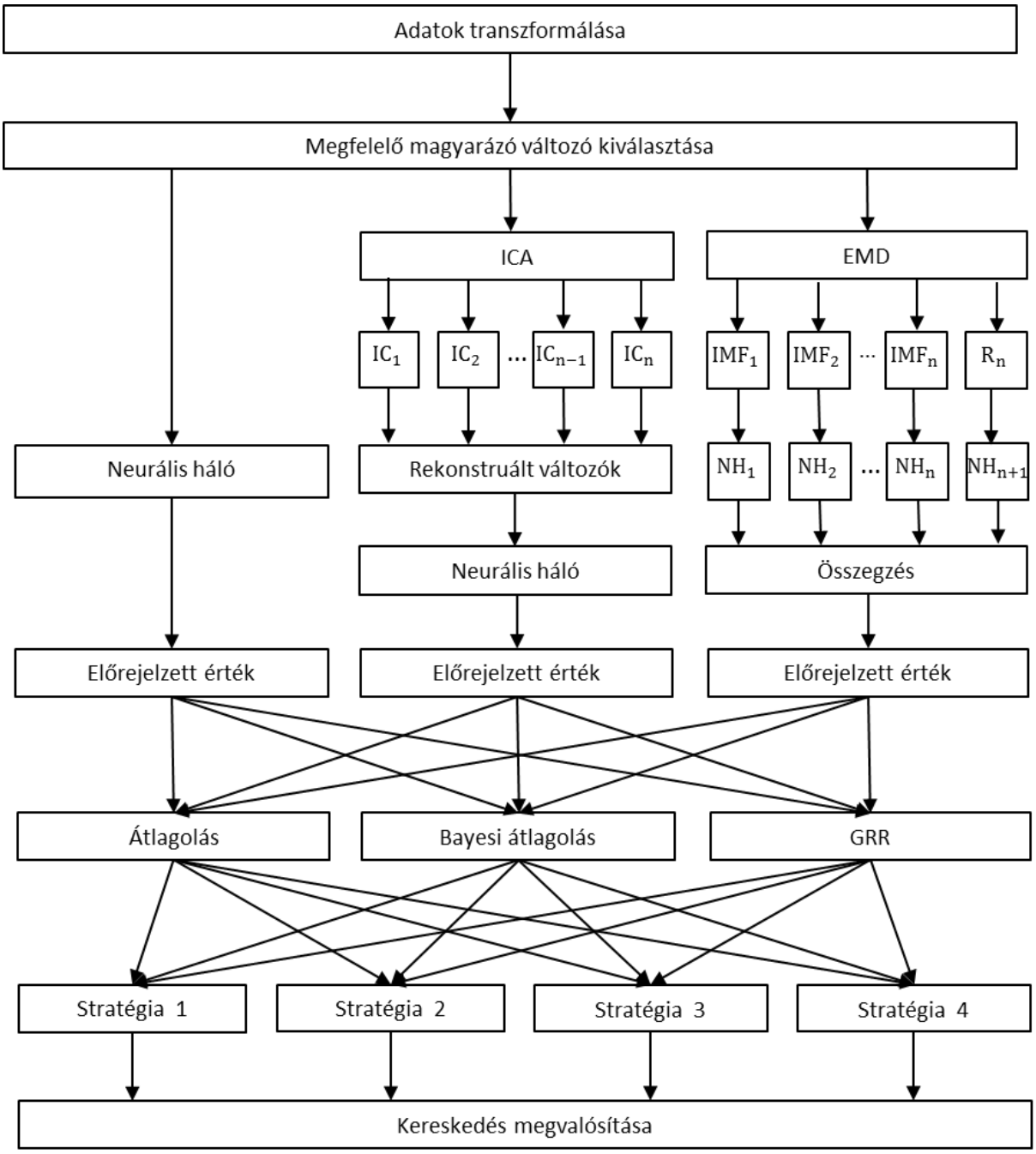
3, Figyelembe veszem az elmozdulások arányát is és az egyes CFD-kbe ennek megfelelő arényban fektetk be: $\mathbf{q_1/(q_1+q_2)}$ illetve $\mathbf{q_2/(q_1+q_2)}$ ahol $\mathbf{q_1}$ és $\mathbf{q_2}$ az előrejelzett hozamok.

4, Mivel a korábbi stratégiák nem vették figyelembe hogy az előrejelzések pontossága a korábbi időszakokban milyen volt, ezért itt a várható érték alapján fektetk be az egyes CFD-kbe: $\mathbf{(p_1*q_1+(1-p_1)*(-q_1))/(p_1*q_1+(1-p_1)*(-q_1)+p_2*q_2+(1-p_2)*(-q_2)))}$ és $\mathbf{(p_2*q_2+(1-p_2)*(-q_2))/(p_1*q_1+(1-p_1)*(-q_1)+p_2*q_2+(1-p_2)*(-q_2)))}$ ahol $\mathbf{p_1}$ és $\mathbf{p_2}$ a korábbi helyes előrejelzett irányok aránya $\mathbf{q_1}$ és $\mathbf{q_2}$ pedig a hozamok.

Az első kivételével az összes modell estén kereskedni fogok minden nap, és a tranzakciós költéget 0,1%-nak veszem.

5. Empirikus elemzés

Dolgozatom empirikus elemzésének folyamatát a 8. ábra mutatja.



8.ábra A dolgozatban bemutatott adatbányászati projekt folyamatábrája (Saját szerkesztés)

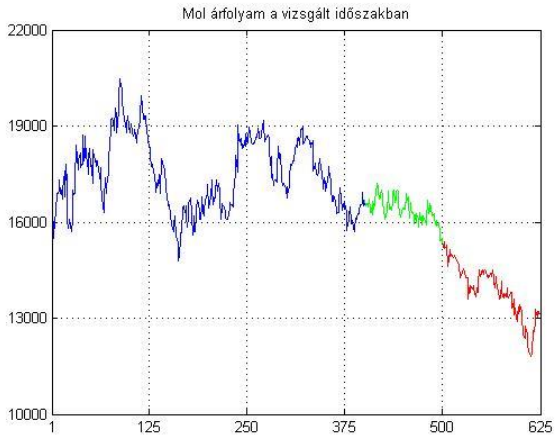
Először elvégzem a változók normalizálást, majd kiválasztom közülük a relevánsakat és azokkal három modellt építek fel (BPN, BPN-ICA, BPN-EMD). Ezután a modellek előrejelzett értékeit három különböző módszer szerint kombinálom (sima átlagolás, bayesi átlagolás, GRR), majd a kapott hozamelőrejelzések segítségével négy befektetési stratégiát

(portfoliót) alkotok a két részvényből (OTP és MOL) és megvizsgálom, hogy ezek felül tudják-e teljesíteni a Markowitz elméleten alapuló statikus portfolió hozamát.

5.1. Adatok és teljesítménykritériumok

Dolgozatomban két budapesti értéktőzsdén forgalmazott részvény felhasználásával alkottam portfoliókat (OTP és MOL), és vizsgált időszaknak a 2011.10.03. és 2014.04.11. közöttit választottam. Mindkét idősort a korábban bemutatottaknak megfelően tanuló, validáló és tesztelő adathalmazokra bontottam, ezek aránya 64%, 16% és 20% lett így a kétésfél éves idősor utolsó fél évén teszteltem mind az előrejelzési módszereket mind a különböző befektetési stratégiákat. Az árfolyamok alakulását a 9-10. ábra mutaja ahol kékkel zölddel és pirossal jelöltem a különböző halmazokat és az 1. táblázat mutaja ezek kezdeti és végpontjait.

9.ábra Az OTP árfolyamának alakulása a vizsgált időszakban (Saját szerkesztés)



10.ábra Az MOL árfolyamának alakulása a vizsgált időszakban (Saját szerkesztés)

Időszak	Adatpontok	Kezdeti dátum	Végső dátum
Teljes időszak	625	2011-10-03	2014-04-11
Tanuló időszak	400	2011-10-03	2013-05-16
Validáló időszak	100	2013-05-17	2013-10-08
Tesztelő időszak	125	2013-10-09	2014-04-11

1.táblázat Az egyes időszakok jellemzői (Saját szerkesztés)

A modellezéshez 8 technikai indikátort választottam, amelyeket széles körben alkalmaznak sok sikerrel, többek között Kara et al., (2011) is. Az indikátorok vizsgált időszakbeli statisztikai tulajdonságait 2. és 15. táblázat (Függelék), míg számítási algoritmusait a 16. táblázat mutatja (Függelék).

	Max	Min	Átlag	Szórás
Súlyozott MA	5302	2835	4061,4	516
Momentum	789	-814	15,3	242,3
Stochastic K%	100	0	53,5	31,3
Stochastic D%	98,8	2,6	53,4	27,2
RSI	88,5	15,2	51,6	15,4
MACD	235,5	-231,4	3,4	76,3
LW R%	0	-100	-47	30,6
A/D Oszcillator	100	0	51,2	28,7

2.táblázat A technikai indikátorok statisztikai jellemzői (Saját szerkesztés)

Az előrejelzési módszerek kiértékelése során három mutatót vettem figyeleme: az eltérések átlagos négyzetösszegének gyökét (RMSE), az átlagos hibaszázalékot (MAPE) és a helyesen eltalált előrejelzések (DA) irányát, ezeket a 17. táblázat definiálja. Mivel korábban említettem hogy magas előrejelzési arány mellett sem biztos hogy felülteljesíti a modellünk a sima „buy-and-hold” stratégiát (például hosszútávon emelkedő piacok esetén), ezért minden egyes előrejelzés esetén megvizsgáltam hogy ha azt felhasználva fektetk be az adott papírba akkor azzal milyen átlagos éves hozamot, és szórást érek el (18. táblázat). Természetesen az első két statisztikai mutatóból az alacsonyabb értékek (az előrejelzett idősor értékei annál közelebb vannak a ténylegeshez minél kisebb a mutatók értéke), míg az előjel-előrejelzési arány és hozam esetén a magasabb értékek az optimálisak.

5.2. A különböző módszerek előrejelzési eredményei

Dolgozatom empirikus részében három adatbányászati modellt alkalmazok majd, azonban mivel mindegyik esetén a modellek gyors konvergenciájához szükséges hogy az inputadatok normalizálva legyenek, ezért első lépésként ezzel kezdem a modellezésemet. Minden egyes változó esetén a következő módszerrel transzformáltam az adatokat a [0,1] intervallumba.

$$x'_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}$$

ahol x_{min} és x_{max} az egyes változók minimuma és maximuma az adott intervallumon.

Következő lépés a relevéns inputváltozók kiválasztása volt amelyhez stepwise regressziót használtam. Mivel csak nyolc inputváltozót vontam be modellembe ezért nem meglepő hogy mindegyiket relevánsnak találta. A módszer alkalmazása tehát akkor célszerűbb amikor jóval több változót szeretnénk használni a modellezés során és szükséges az inputváltozók által kifeszített tér dimenziójának redukálása.

Mindhárom adatbányászati modellem esetén a Matlab programot és annak különböző Toolbox-ait (Neural Network Toolbox, Statistical Toolbox) és package-eit (FastICA package, EMD package) használtam. Ennek oka egyrészt az volt, hogy a módszerek jól le vannak programozva, és könnyen parametrizálhatóak a különböző idősorokhoz, másrészt pedig a program nagyon gyorsan képes a műveleteket elvégezni, így az optimális paraméterek beállításhoz szükséges nagyszámú modell futtatása nem okoz gondot.

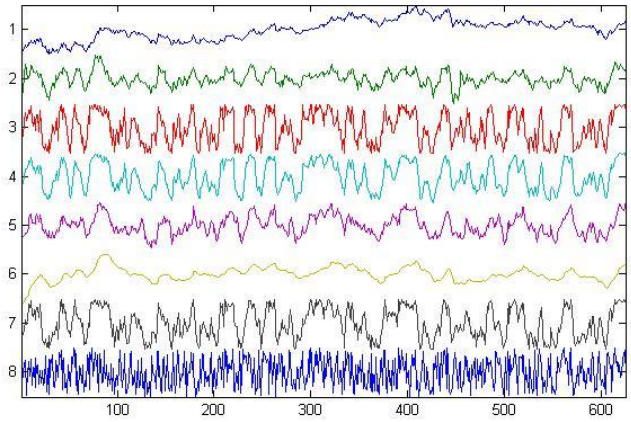
Első lépésben az egyik legnépszerűbb neurális hálót (backpropagation neural network BPN) használtam modellezésem során. A megfelelő paraméterek (rejtett rétegben lévő neuronok száma, tanulási ráta) kiválasztáshoz a grid search eljárást használtam. A hálózat input rétege 8 neuronból állt (a magyarázó változók számának megfelelően), míg a köztes rétegben a 11, 12, 13, 14 neuronszámú hálózatokat teszteltem. A hálózatnak egy kimenete volt: a részvények hozama. Lu (2010) tanulmánya alapján alacsony tanulási ráták (0,01, 0,02, 0,03, 0,04, 0,05) mellett teszteltem a modelleket a tanulási folyamat alatt. Konvergenciakritériumként azt a szabályt alkalmaztam, hogy a tanulási folyamat leáll, ha az RMSE mutató kisebb lesz mint 0,0001 vagy eléri az 1000-ik iterációt. Azt a hálózati topológiát választottam optimálisnak amely esetén a teszthalmazon a legkisebb az RMSE. A 3. táblázat mutatja a neurális hálózat különböző paraméterei esetén a teszthalamzon mért

teljesítményt, amely alapján a későbbiekben validációs halmazon történő a modellezés során 8-12-1 es topológiával és 0,05 tanulási rátával rendelkező hálózatot használtam.

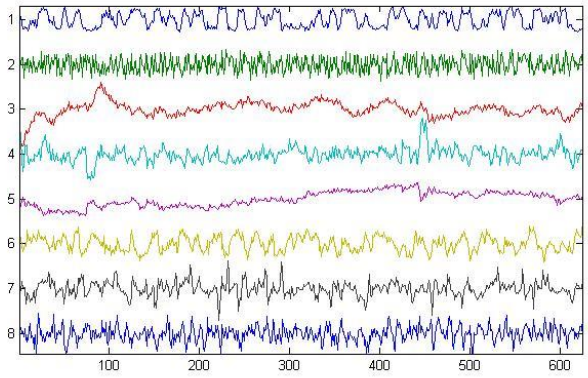
Rejtett rétegben lévő neuronok száma	Tanulási ráta	Validációs RMSE
11	0,01	0,124111
	0,02	0,120873
	0,03	0,119689
	0,04	0,119021
	0,05	0,118578
12	0,01	0,120424
	0,02	0,117532
	0,03	0,116893
	0,04	0,116581
	0,05	**0,116369**
13	0,01	0,124840
	0,02	0,123034
	0,03	0,121980
	0,04	0,121219
	0,05	0,120619
14	0,01	0,124489
	0,02	0,120798
	0,03	0,119771
	0,04	0,119247
	0,05	0,118872

3.táblázat Különböző paraméterű BPN hálózatok hibája a teszthalmazon (Saját szerkesztés)

Mivel a pénzügyi idősorokra jellemző hogy magas a zaj/jel arány ezért második modellemben a BPN háló használata előtt a független komponenselemzéssel kiszűrtem az inputváltozókból a zajt. Ehhez szükséges volt egyrészt a független-komponensek (IC-k) előállítása, majd a TnA algoritmus segítségével a zaj komponens definiálása. Az OTP esetén a 11. és a 12. ábra mutatja az eredeti inputváltozókat (már normalizált formában) és megfelelő független-komponenseket.



11.ábra Az OTP modellezéséhez használt inputváltozók (Saját szerkesztés)



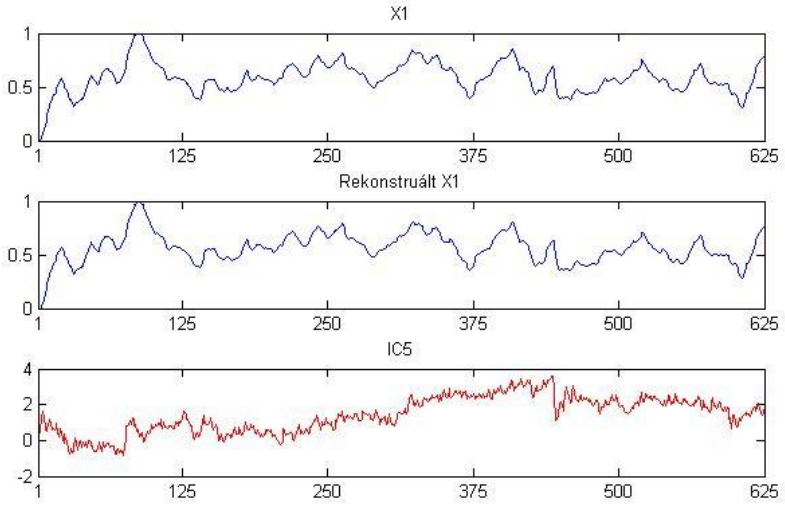
12.ábra Az inputváltozókból képzett független-komponensek (Saját szerkesztés)

A TnA algoritmus során egyesével elhagyjuk az egyes IC-ket, majd rekonsturáljuk az inputmátrixot, és megvizsgáljuk hogy ez mennyire tér el az eredetitől. Az eltérést az RHD mutatóval tudjuk mérni. Mivel esetünkben 8 input változó van ezért 7-szet kell ezt a műveletetet elvégeznünk hogy megtaláljuk a zaj komponenst, ezek RHD értékeit mutatja a 4. táblázat.

Főkomponensek	RHD
IC1, IC2, IC3, IC4, IC5, IC6, IC7	4,3674
IC1, IC2, IC3, IC4, IC5, IC6, IC8	3,6260
IC1, IC2, IC3, IC4, IC5, IC7, IC8	4,4830
IC1, IC2, IC3, IC4, IC6, IC7, IC8	**2,4118**
IC1, IC2, IC3, IC5, IC6, IC7, IC8	3,7873
IC1, IC2, IC4, IC5, IC6, IC7, IC8	3,9655
IC1, IC3, IC4, IC5, IC6, IC7, IC8	7,1473
IC2, IC3, IC4, IC5, IC6, IC7, IC8	7,7748

4.táblázat Különböző rekonstruált inputmátrixok RHD értékei (Saját szerkesztés)

A táblázat alapján megállíptható, hogy az ötödik komponens a zaj, ezt, az enélkül rekonsturált x_1 változót és az eredetit mutatja az 13. ábra. Az ábrán lehet látni hogy az eredeti és a rekonstruált változó idősora nagyon hasonló, a kettő különbsége csak az hogy utóbbiból a zaj komponens el lett távolítva, így várhatóan az előrejelzések során ennek felhasználása többletprofitot eredményez majd.



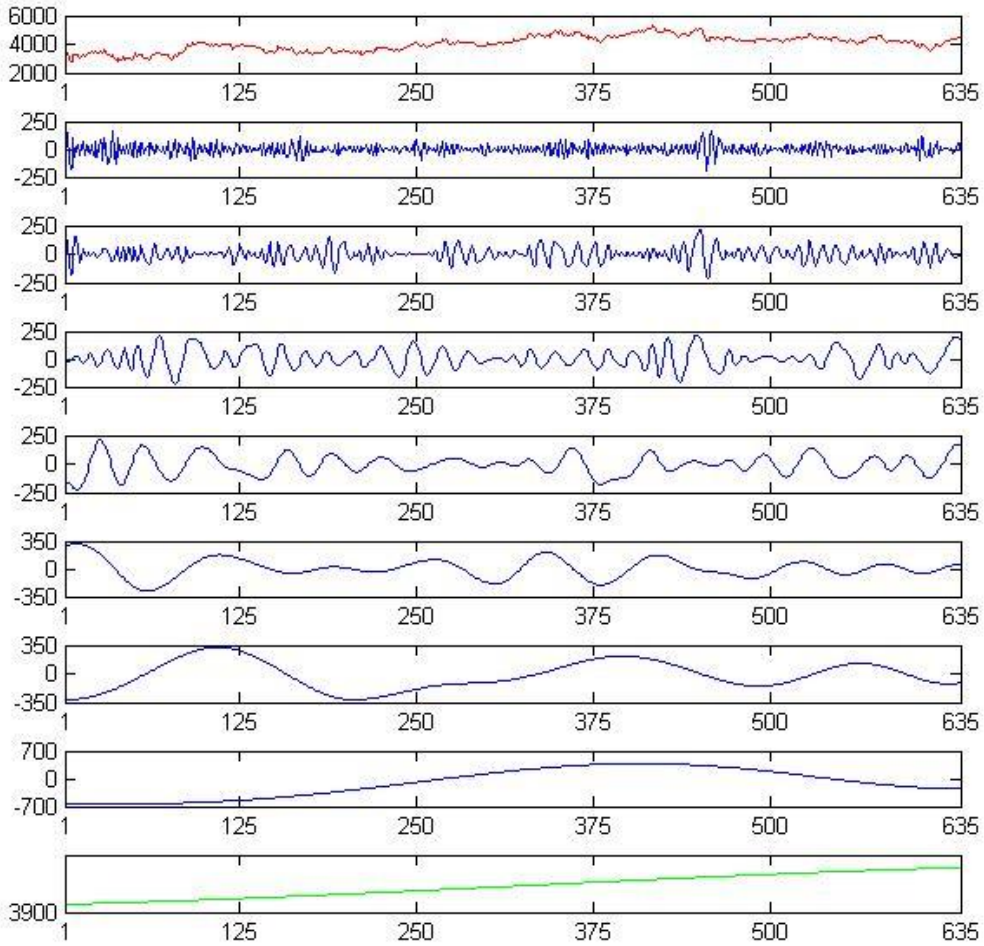
13.ábra Az eredeti és a rekonstruált x_1 változó illetve a zaj komponens (Saját szerkesztés)

A modell második lépseként rekonstruált változók felhasználásával BPN hálózatot építünk. Az optimális paraméter kiválasztása teljesen hasonlóan működik a korábban bemutatotthoz, az 5. táblázat alapján az ICA-BPN modell esetén is a 8-12-1 topológiájú hálózat lesz az optimális.

Rejtett rétegben lévő neuronok száma	Tanulási ráta	Validációs RMSE
11	0,01	0,122957
	0,02	0,121980
	0,03	0,121522
	0,04	0,121176
	0,05	0,120894
12	0,01	0,120471
	0,02	0,119321
	0,03	0,119095
	0,04	0,118994
	0,05	**0,118908**
13	0,01	0,125093
	0,02	0,124105
	0,03	0,123608
	0,04	0,123250
	0,05	0,122951
14	0,01	0,122215
	0,02	0,120560
	0,03	0,120102
	0,04	0,119888
	0,05	0,119794

5.táblázat Különböző paraméterű ICA-BPN hálózatok hibája a teszthalmazon (Saját szerkesztés)

A harmadik módszer amit dolgozatomban használni fogok a „divide-and-conquer” elven alapszik. A tőzsdei idősorok komplex dinamikája miatt az eredeti idősort az EMD módszer segítségével IMF-ekre bontom fel, és ezeket külön külön előrejelzve majd összeadva kapom meg az előrejelzett értékét. AZ ICA-hoz hasonlóan ezzel is tudjuk az idősorok zajszűrést elvégezni ugyanakkor itt az nem az inputadatokból történik, hanem magából az előrejelzendő részvény idősorából. Több tanulmányhoz hasonlóan (Yu et al., 2008;Cheng &Wei, 2014; Lin et al., 2012) ennél a módszernél én is az árfolyamokat jeleztem előre, az OTP árfolyamának empirikus alapú dekompozícióját mutatja a 14. ábra.



14.ábra Az OTP árfolyamainakempirikus alapú dekompoziciója (Saját szerkesztés)

Legfelső sorban pirossal jeleztem az eredei idősort majd kékekkel a különböző egyre kisebb frekvenciájú IMF-eket (IMF1, IMF2, …, IMF8) és végül zölddel a trendnek megfeleltethető reziduumot. A módszer második lépésenként minden egyes IMF-et különböző paraméterű neurális hálókkal előrejeleztem majd a kapott értékeket aggregálva kaptam meg az idősor előrejelzett értékét. Mivel ebben az esetben végső soron 8 idősort kellett előrejelznem és ezekhez meghatároznom az optimális inputok számát (hány késleltetést alkalmazzak a NAR modellben) ezért ez a korábbi két modellhez képest jóval komlexebb és időigényesebb folyamat volt. A probléma megoldhatósága érdekében a késleltetések számát minden egyes IMF esetén 10-ben határoztam meg Mingming & Jinliang (2012) alapján és így csak az optimális neuronszámot és tanulási rátát kellett megkeresnem. Ezeket a különböző IMF-ek esetén a 6. táblázat mutatja.

IMF	Neuronok száma	Tanulási ráta
1	12	0,05
2	12	0,05
3	12	0,05
4	12	0,025
5	12	0,025
6	12	0,025
7	13	0,025
8	13	0,025

6.táblázat Különböző IMF-ek optimális paraméterei (Saját szerkesztés)

A három modell optimális paramétereinek megtalálása után a validációs időszakra való előrejelzéshez használtam őket. A MOL részvény modellezése során kapott optimális paramétereket a xx.-xx. táblázatok (Függelék), míg az validációs halmazon futtatott előrejelzések statisztikai adatait a 7. (OTP) és 8. (OTP) táblázatok mutatják.

Modell	RMSE	MAPE (%)	DA (%)
BPN	0,018864	113,38	**61,6**
ICA-BPN	0,018738	107,79	**60,8**
EMD-BPN	0,026672	292,47	**56,8**

7.táblázat Különböző módszerek teljesítménye a validációs halmazon (OTP) (Saját szerkesztés)

Modell	RMSE	MAPE (%)	DA (%)
BPN	0,015863	133,65	**56**
ICA-BPN	0,015910	127,60	**58,4**
EMD-BPN	0,014568	162,71	**62,4**

8.táblázat Különböző módszerek teljesítménye a validációs halmazon (MOL) (Saját szerkesztés)

A táblázatok alapján lehet látni hogy a feljettebb hybrid módszerek előjel-előrejelzési aránya nem (OTP) vagy csak minimálisan (MOL) jobb a sima BPN modellnél, azonban később érdemes lesz azt is majd megnézni hogy elért profit szempontjából felülmúlják-e az első modellt.

Előtte azonban még megvizsgáltam hogy a három módszert kombinálva javulnak-e az előrejelzései eredmények. Ahogy korábban említettem a kombinálás segítségével ki tudjuk

küszöbölni az egyes módszerek hártányait, ezáltal jobb előrejelzést és magasabb profitot tudunk elérni. A három módszer (sima átlag, bayesi átlag, GRR) háromfajta kombinálásával kapott előrejelzés eredményeit foglalja össze a 9. és 10. táblázat.

Modell	RMSE	MAPE (%)	DA (%)
Átlag	0,018854	144,82	**64,8**
Bayesi Átlag	0,018733	107,87	**60,8**
GRR	0,019087	151,21	**61,6**

9.táblázat A három módszer kombinálásával kapott eredmények (OTP) (Saját szerkesztés)

Modell	RMSE	MAPE (%)	DA (%)
Átlag	0,014450	111,90	**61,6**
Bayesian átlag	0,014568	162,71	**62,4**
GRR	0,014158	134,02	**60,8**

10.táblázat A három módszer kombinálásával kapott eredmények (MOL) (Saját szerkesztés)

A kombinációk eredményeit vizsgálva egy furcsaságot vehetünk észre: a bayesi átlag MOL részvény esetén ugyanazt az eredményt adja mint az EMD-BPN modell. Ennek oka hogy a bayesi átlagolás esetén ha egy módszer nagyon jó eredmény ér el a tanuló és a teszthalmazon akkor azt felülsúlyozza az átlagolás során, és ebben az esetben az EMD-BPN annyival jobb előrejelést mutatott a két halmazon hogy csak azt. A három adatbányászati és a három kombinációs modell validációs halmazon elért profitjait mutaja a 11. és 12. táblázat illetve a 15. és 16. ábra.

	Buy and hold	BPN	ICA-BPN	ICA-EMD	Átlag	Bayesi Á.	GRR
Éves hozam	**2,50%**	**36,46%**	**71,53%**	**64,01%**	**124,27%**	**62,71%**	**85,16%**
Éves volatilitás	30,63%	30,22%	29,68%	29,92%	28,73%	29,81%	29,52%

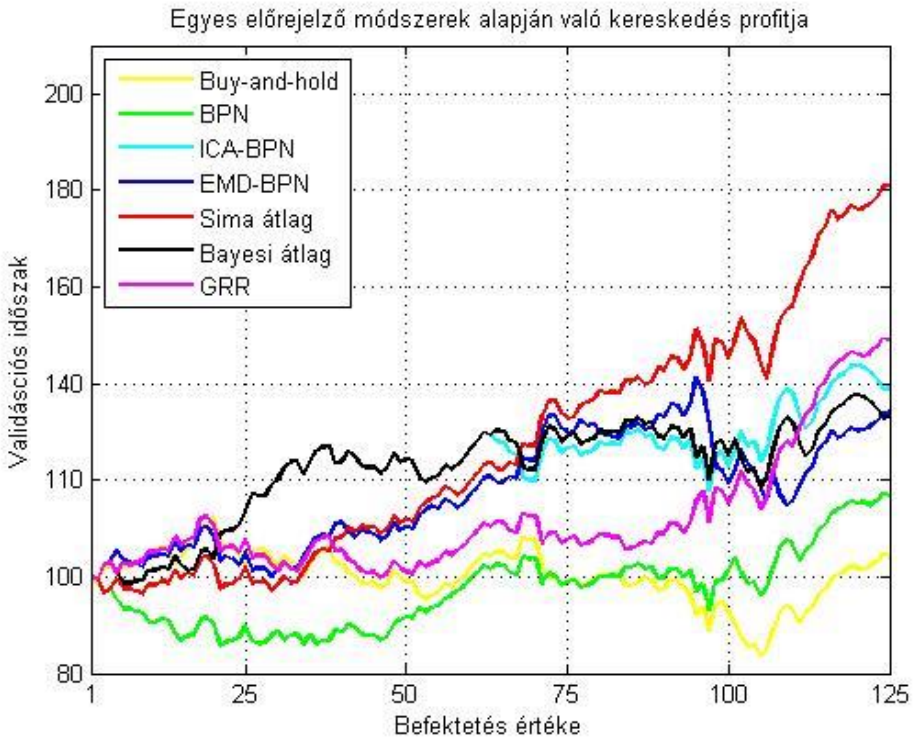
11.táblázat A 6 modell által generált profit a validációs halmazon (OTP) (Saját szerkesztés)

	Buy and hold	BPN	ICA-BPN	ICA-EMD	Átlag	Bayesi Á.	GRR
Éves hozam	**-34,79%**	**48,40%**	**56,31%**	**83,09%**	**115,36%**	**83,09%**	**114,83%**
Éves volatilitás	25,91%	24,61%	24,56%	24,03%	23,29%	24,03%	23,30%

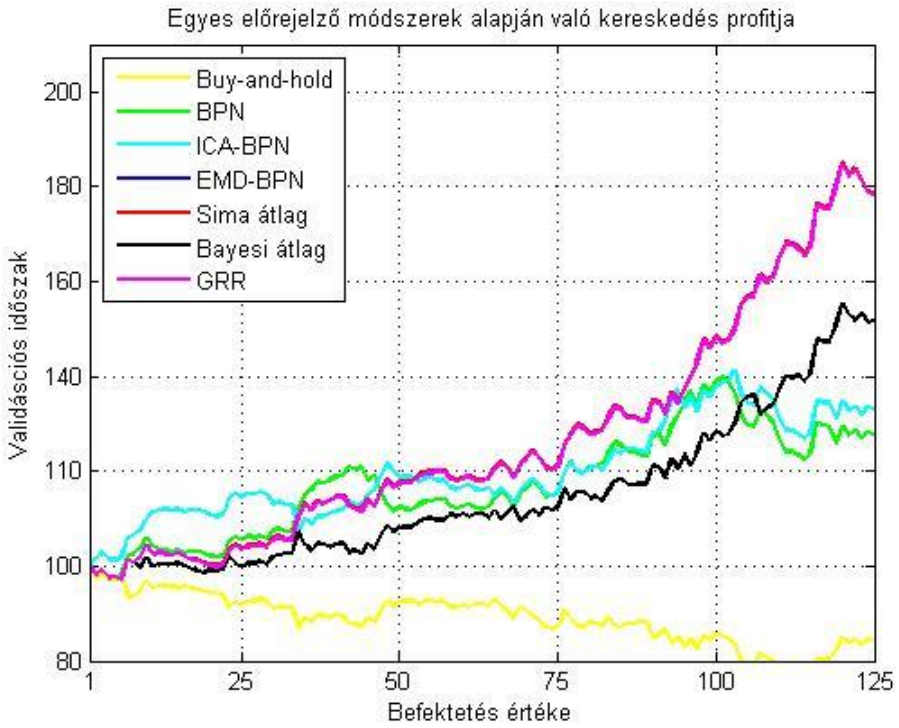
12.táblázat A 6 modell által generált profit a validációs halmazon (MOL) (Saját szerkesztés)

A táblázatokat vizsgálva egyrészt megfigyelhetjük hogy habár az előjel-előrejelzési arány a 2 komplexebb adatbányászati modell esetén nem volt sokkal jobb mint a sima neurális hálónál, azonban profit szempontjából messze felülmúlják azt. Másrészt mindhátom modell jobban teljesít mint a „buy-and-hold” stratégia főleg a MOL részvény esetén ahol az adott időszak alatt a részvény nagy zuhanása miatt ezzel a stratégiáva 35%-ot veszítettünk

volna a befektetett tőkénkből. Emellett azt is feltünő, hogy a három kombinációs módszerből profit szempontjából a sima átlagolás teljesít a legjobban. Ez elsőre furcsának tűnhet mivel ez a legkevésbé szofisztikált átlagolási módszer, azonban ha belegondolunk hogy a másik kettő a tanuló és teszthalmazon elért hibák alapján súlyozza a modelleket, és itt a profit szempontjából jobban teljesítő modellek (ICA-BPN és ICA-EMD) a sima neurális hálóhoz képest rosszabbul teljesítettek, így ezeket kisebb súllyal átlagolja. Az ok hogy emiatt alacsonyabb profit-ot eredményeznek ezek a kombinációk az az, hogy a két módszer (ICA-BPN és EMD-BPN) a validációs időszakon közel ugyanolyan jól jelez előre mint a teszt és tanuló időszakon, míg a sima neurális háló rosszabbul. Így amikor nagyobb súllyal szerepelnek a kombinációban (sima átlagolás) akkor az több profitot eredményez. Érdemes lenne a későbbi kutatások során megivzsgálni hogy mi történne, ha nem RMSE hanem profit alapján történne a másik két kombinációs módszer súlyainak választása.



15.ábra Az egyes adatbányászati és kombinációs modellek alkalmazásával elérhető profit a validációs halmazon (OTP) (Saját szerkesztés)



16.ábra A négy portfolióallokációs stratégia alkalmazásával elérhető profit a validációs halmazon (Saját szerkesztés)

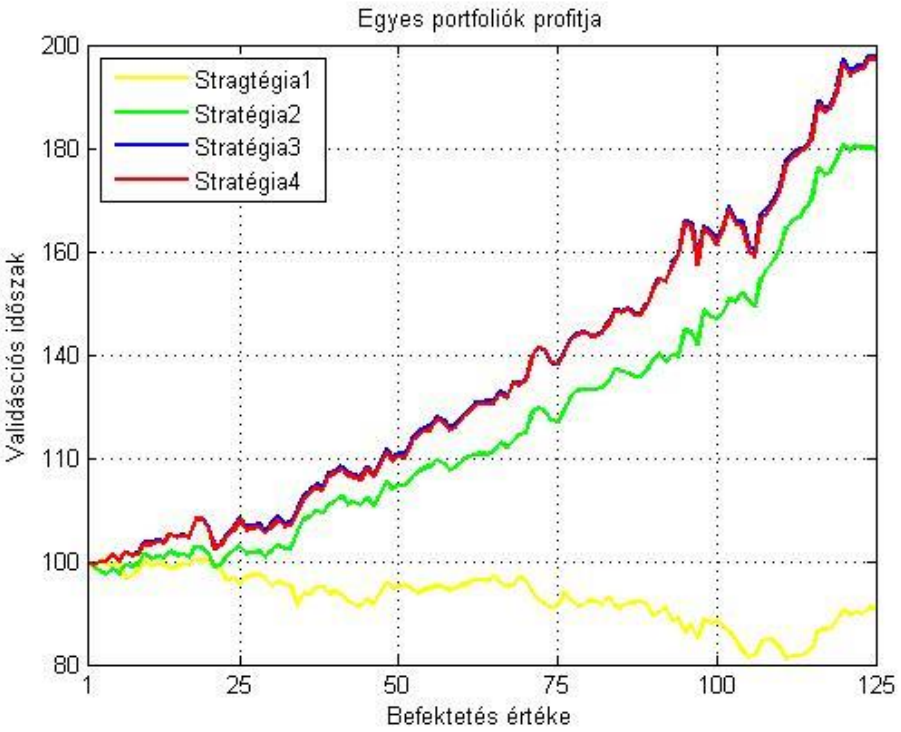
Az ábrák és táblázatok alapján látható, hogy az előrejelzési módszerek segítségével magas profitot tudunk elérni, ezért érdemes megvizsgálni hogy a két részvényből álló különböző portfoliók hogy teljesítenek a validációs halmazon.

5.3. Portfoliokezelés adatbányászati módszerekkel

A korábbi fejezetben bemutatott négy portfolióallokációs stratégiának az eredményességét mutatja a 17. ábra és a 13. táblázat.

	Stratégia 1	Stratégia 2	Stratégia 3	Stratégia 4
Éves hozam	**-21,78%**	**118,87%**	**140,50%**	**140,62%**
Éves volatilitás	22,73%	19,16%	22,94%	23,36%

13.táblázat A 4 portfolióallokációs stratégia által generált profit a validációs halmazon (Saját szerkesztés)



17.ábra Az egyes adatbányászati és kombinációs modellek alkalmazásával elérhető profit a validációs halmazon (MOL) (Saját szerkesztés)

Mind az ábráról, mind a táblázatból jól látható hogy az adatbányászati módszerekkel való előrejelzést felhasználó portfolió-allokációs stratégiák messze felülmúlják a Markowitz által javasolt portfoliót. Utóbbi azért ér el negatív hozamot, mivel a tanuló és teszt időszak alapján számolt várható értékek, varanciák, és kovariancia alapján az optimális súlya a nagy validációs időszak alatt veszteséget elérő MOL-nak 72%, míg az pozitív profitot elérő OTP-nek csak 28%. Ebből is látszik, hogy ilyen hosszú időszak múltbeli hozamait és szórásait előrejelzésre csak komoly fenntartásokkal használhatjuk. A pénzügyi/tőzsdei idősorok nemlinearitása és nem-stacioner jellege miatt érdemesebb inkább előrejelző módszereket, azon belül is adatbányászati modelleket használni. Az ilyen módszerek közül a legnagyobb profitot akkor értük el amikor az előrejelzések múltbeli reljesítményét is figyelembe vettük a részvények portfolión belüli súlyainak meghatározásához (Stratégia 4).

6. További kutatási lehetőségek, a módszer alkalmazásának kihívásai

Ahogy az előző fejezet eredményei alapján láttuk az adatbányászati technikák segítségével megvalósuló aktív portfoliókezelés felül tudja múlni a „buy and hold” stratégiát, ugyanakkor a modellalkalmazás nem egy egyszerű, mivel megvannak a maga nehézségei, és

korlátai. Ezek közé tartozik az optimális módszerek, paraméterek kiválasztása, illetve bizonyos időszakonként (akár naponta is) a modellek újrakalibrálása, amely rendkívül időigényes folyamat és nagy körültekintést követel meg. Véleményem szerint itt igazolódik a hatékony piacok egyik állítása - igaz másként, mit ahogy értelmezni szokták a kutatók -, ugyanis a módszer segítségével az átlagoshoz képest többlethozamot tudunk elérni, ugyanakkor ez megköveteli az az információt/tudástöbbletet, amit a modellek kifejlesztése, alkalmazása során használunk, és feltételezésem szerint ezt nem miden piaci szereplő birtokolja, hanem azoknak csak egy része. Az információ itt tehát nem az elérhető adatokra vonatkozik, hanem a módszerek alkalmazásainak lehetőségeire és korlátaira, így a modellek folytonos fejlesztésére van szükség, hogy mindig a piac átlagát reprezentáló tudáshoz képest egy vagy két lépéssel előrébb járjunk.

Emiatt a következőekben pár továbblépési lehetőséget, kutatási irányt szeretnék bemutatni. Természetesen érdekes lehet megvizsgálni hogy a különböző értékapapírokra ugyanazok a módszerek adják-e a legpontosabb előrejelzést, illetve ha nem, akkor annak a részvénynek milyen jellemzőjében való különbség az oka. Emellett érdemes figyelembe venni és alkalmazni a dolgozatban kevésbé említett adatbányászati módszereket a genetikus algoritmusok használatától kezdve, a döntési fákon át a szöveges adatbányászatig. Utóbbi az elmúlt 2–3 év leggyorsabban fejlődő területe, amely során a piacon megjelenő híreket automatizálva elemzik és megállapítják azok hatását az egyes értékpapírokra (Hagenau et al., 2013). Ezenkívül érdemes megvizsgálni, hogy más frekvenciák esetén (napi ehelyett óránként/percenkénti előrejelzés) milyen eredményeket kapunk, javít-e a portfolió hozamán a gyakoribb kereskedés. Fontos kutatási terület-irány az is, hogy mik azok a részvények, amelyeket eredményesen lehet előrejelzni akár csak az egyes tőzsdéken belül akár az egész világon tekintve. Erre a 90-es évek végén kezdték el alkalmazni a Hurst-exponenst (Lin et al., 2009; Eom et al., 2008) tőzsdei körökben ugyanis ennek a változónak az értéke ki tudja fejezni az egyes idősorok előrejelzhetőségét (hosszú-távú memóriáját). Ha a mutató értéke nem 0,5 körül van akkor alkalmazhatóak az előrejelző módszerek mivel a folyamatban valamilyen függőségi viszony van az egymást követő értékek között. Ugyan az ezzel kapcsolatos kutatások száma még nem nagy, véleményem szerint ez a következő években meg fog változni, hiszen alkalmazásával megkereshetőek lesznek a részvények, amelyeknél érdemes az előrejelzéssel próbálkozni ezzel meggyorsítva az aktív portfoliókezeklés folyamatát. A probléma még komplexebbé válik, ha a részvényeken kívül más pénzügyi

eszközök például devizák (Sermpinis et al., 2013), kötvények (Tay & Cao, 2001) vagy opciók (Sheu & Wei, 2011) árfolyamát is megpróbáljuk előrejelzni, hiszen ezen termékek idősor jellegében eltérhet a részvényekétől (devizák esetén a zaj/jel arány véleményem szerint magasabb lehet), így más-más módszerek alkalmazása lehet optimális ezen papírok esetén.

Kereskedés szemponjtából fontos lehet vizsgálni hogy milyen profitot tudunk elérni magasabb tőkeáttétel esetén, hiszen erre rendkívül sok piacon van lehetőség, van ahol akár 400-szoros tőkeáttételes pozíciókat is fel tudunk venni. Sermpinis et al., (2012) bemutatott erre egy módszert, ahol az árfolyamok mellett a papírok volatilitását is előrejelezte, és aszerint határozta meg a tőkeáttétel mértékét, hogy ez mekkora volt (magas tőkeáttétel alacsony volatilitás esetén, alacsony pedig magas volatilitás esetén). Azonban ez szinte egyedi eset, a kutatások nagy rész nem foglalkozik ezzel, így érdemes lehet alaposabban megvizsgálni egyrészt a volatilitást előrejelzésének lehetőségét, illetve az ez alapján felállított tőkeáttételi szabályok érvényességét. Az általam definiált portfoliók is mutatják, hogy az optimális potfoliókeresésének relevanciája van, ezt tükrözi az elmúlt években megjelent jelentős számú kutatás is (Freitas et al., 2009; Chen et al., 2010). Ugyanakkor nemcsak azt a stratégiát lehet definiálni, hogy előrejelzés alapján, ha nő a papír értéke akkkor megvesszük, ha csökken akkor eladjuk, hanem ennél kifinomultabbakat is, mint például a gyakran használt pair trading stratégiát (Huck, 2010). Ebben az esetben nem egyedi részvények árfolyammozgására spekulálunk, hanem két részvény eggyüttmozgásának erősségére. Ez a 80-as években nagyon elterjedt volt a befektetési alapok körében, így hasznos lehet megvizsgálni hogy ezt hogyan tudja támogatni egy adatbányászati alapokon működő rendszer.

Természetesen a dolgot meglehet közelíteni erőforrásszempontból is, mivel ahogy említettem, egyes modellek számítási igánye jelentős lehet. Ha figyelembe vesszük, hogy akár nagyon gyorsan (5 percenként) kell előrejeleznünk több száz részvény értékét különböző modellekkel, majd megtalálni az optimális portfoliósúlyokat, akkor elérkezünk a Big Data világába, amely az adatbányászat legújabb és leggyorsabban fejlődő területe. Nagyon nagy mennyiségű és változékony adat kezelésére nem biztos, hogy azok a a módszerek a legjobbak, amik kis adathalmazon jól működnek, így egy újabb kíhívást írhatunk fel a módszerek alkalmazási lehetőségei közé.

A felsorolt indokok alapján lehet látni hogy ugyan az adatbányászattal jelentős többlethozamot lehet elérni a hagyomásnyos befektetési stratégiákhoz képest, azonban

ennek kivieléze egy rendkívül komplex probléma, így komoly erőforrásokat és szakértelmet követel meg. Az elért eredményeket tehát nem „ingyen” adják a tőzsdén, hanem komoly áldotatokat kell hozni érte.

7. Konkluzió

A befektők mellett az elmúlt évizedekben a kutatók érdeklődését is felkeltette a pénzügyi/tőzsdei idősorok előrejelzésének módszertana, és a 80-as évektől kezdve a hagyományos statisztikai/ökonometriai modelleket felváltották az adatbányászati módszerek, amelyeket jobban lehet használni nemlineáris, nemstacioner idősorok esetén. Az elúlt 20-25 évben rengeteg -korábban a mérnöki világban alkalmazott- adatbányászati módszert keztek el használni tőzsdei idősorok előrejelzésére, egyre újabb és szofisztikáltabb modellek jelentek meg a szakirodalomban és a piaci alaklamazások során is. Dolgozatom célja ezért a különböző adatbányászati modellek aktív portfoliókezelés szempontjából való felhasználhatóságának bemutatása volt. Olyan árfolyamelőrejelzésen alapuló portfolió-allokációs stratégia kidolgozását szerettem volna megvalósítani, amely trzancakciós költségek mellett is felül tudja múlni a hagyományos Markovitz modellen alapuló stratégiát.

Dolgozatomat eleján részletesen kitértem arra hogy miért érdemesebb pénzügyi idősorok esetén statisztikai módszerek helyett adatbányászati modelleket használni, bemutattam ezek közül a leggyakrabban alkalmazottakat azok előnyeivel és hártányaival együtt, és kitértem arra is hogy milyen az elmúlt évtizedekben bevezetett új módszerekkel lehet az idősorok magas zaj/jel arányát, és komplexitását kezelni. Az általam legjobbnak tartott modellek segítségével előrejelzést végeztem a budapesti értéktőzsde két részvényének árfolyamán és ezeket felhasználva több portfolió-allkoációs stratégiát alkottam.

A dolgozat elején kitűzött célt úgy érzem sikerült teljesíteni mivel az előrejelzéseimen alapuló stratégiák messze felülmúlták a hagyományos portfolió-allokációt és a validációs halmazon évi 120-140%-os hozamot is el tudtak érni. Azonban arra is rámutattam, hogy ezek alkalmazása komplex és sok szakértelmes igénylő feladat, sok nehézséget és problémát kell leküzdenünk ha a való életben is szeretnénk használni a kereskedés során. Mivel dolgozatomban a teljes adatbányászati folyamatot bemutattam a szükséges input változók kiválasztásától a használható adatbányászati módszerek definiálásán át egészen az optimális portfolió kialaktásáig, így dolgozatom „utikönyvként” alkalmazható a felmerülő problémák/nehézségek leküzdésére amikor élesben használjuk a modelleket. Összességében úgy érzem, hogy egy értékes alkotást tettem le az olvasó asztalára, amely egyrész támpontot nyújthat későbbi kutatások elvégzéséhez, másrészt a módszerek gyakorlati alkalmazásához is.

8. Hivatkozások

Armano, G., Marchesi, M. & Murru, A. (2004). A hybrid genetic-neural architecture for stock indexes forecasting. Information Sciences, 170(1), 3–33.

Armstrong JS. (1989) Combining forecasts: the end of the beginning or the beginning of the end? International Journal of Forecasting, 5, 585–5888.

Asadi, S., Hadavandi, E., Mehmanpazir, F. & Nakhostin, M. M. (2012). Hybridization of evolutionary Levenberg–Marquardt neural networks and datapre-processing for stock market prediction. Knowledge-Based Systems, 35, 245-258.

Atsalakis, G. S., Valavanis, K.P. (2009). Surveying stock market forecasting techniques-Part II. Soft computing merthods. Expert Systems with Applications, 36(3), 5932-5941.

Baban, S. (2008). Design and Implementation of a Scheduling Algorithm for the IEEE 802.16e (Mobile WiMAX) Network. Master's thesis.

Back, A. & Weigend, A. (1997). Discovering structure in finance using independent component analysis. In Proceeding of fifth international conference on neural networks in capital market, 15–17.

Bell, A. J. & Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. Neural Computation, 7, 1129–1159.

Barnes, M. B., Rimmer, R. J. & Ting, K. M. (2000). A study of techniques for mining data from the Australian stock exchange, In: Proceedings of the Fourth World Multi-conference on Systemics. Cybernetics and Informatics, 8(2), 52–57.

Beckmann, C. F. & Smith, S. M. (2004). Probabilistic independent componentanalysis for functional magnetic resonance imaging. IEEE Transactions on Medical Imaging, 23(2), 137–152.

Cao, L.J. & Tay, F.E.H. (2001). Financial forecasting using support vector machines, Neural Computing & Applications, 10, 184–192.

Cao, L. J., & Chong, W. K. (2002). Feature extraction in support vector machine: a comparison of PCA, XPCA and ICA, In Proceedings of the ninth international conference on neural information, 1001–1005.

Cao, L., & Gu, Q. (2002). Dynamic support vector machines for non-stationary time series forecasting. Intelligent Data Analysis, 6(1), 67–83.

Cao, L.J. (2003). Support vector machines experts for time series forecasting. Neurocomputing, 51, 321–339.

Cao, Q., & Parry, M. E. (2009). Neural network earnings per share forecasting models: A comparison of backward propagation and the genetic algorithm. Decision Support Systems, 47(1), 32–41.

Chan, Y.L., Stock, J.H. & Watson, M.W. (1999). A dynamic factor model framework for forecast combination. Spanish Economic Review, 1(2), 91–121.

Chang, P-C., Liu, C-H., Lin, J-L., Fan, C-Y. & Ng, C. S. P. (2009). A neural network with a case based dynamic window for stock trading prediction. Expert Systems with Applications, 36(3), 6889–6898.

Chang, T-S. (2011). A comparative study of artificial neural networks, and decision trees for digital game content stocks price prediction. Expert Systems with Aplications, 38(12), 14846-14851.

Chauvin, Y. & Rumelhart, D. E. (1995). Backpropagation: Theory, architectures, and applications. New Jersey: Lawrence Erlbaum associates.

Chaturvedi, A. & Chandra, S. (2004). A neural stock price predictor using quantitative data. Proceedings of the Sixth International Conference on Information Integration and Web-Based Applications Services, 27–29.

Chavarnakul, T. & Enke, D. (2008). Intelligent technical analysis-based equivolume charting for stock trading using neural networks. Expert Systems with Applications, 34(2), 1004–1017.

Chen, A-S., Leung, M. T. & Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan Stock Index. Computers & Operations Research, 30(6), 901–923.

Chen, Y., Mabu, S., Hirasawa, K. (2010). A model of portfolio optimalization using time adapting genetic network programming. Computers & Operations Research, 37(10), 1697-1707.

Chen, C.F., Lai, M.C. & Yeh, C.C., (2012). Forecasting tourism demand based on empirical mode decomposition and neural network. Knowledge-Based System 26, 281–287.

Cheng, C-H. & Wei L-Y. (2014). A novel time-series model based on emprical mode decomposition for forecasting TAIEX. Enomic Modelling, 36, 136-141.

Cheung, Y. M. & Xu, L. (2001). Independent component ordering in ICA time series analysis. Neurocomputing, 41(1-4), 145–152.

Chun S-H., Kim S.H. (2004). Data mining for financial prediction and trading: application to single and multiple markets. Expert Systems with Applications, 26 (2), 131–139.

Constantinou, E., Georgiades. R., Kazandjian, A. & Kouretas, G. P. (2006). Regime switching and artificial neural network forecasting of the Cyprus Stock Exchange daily returns. International Journal of Finance and Economics.

Dai, W., Wu, J-Y. & Lu, C-J. (2012). Combining nonlinear independent component analysis and neural network for the prediction of Asian stock market indexes. Expert Systems with Application, 39(4), 4444-4452

David, V. & Sanchez, A. (2002). Frontiers of research in BSS/ICA. Neruocomputing, 49(1), 7–23.

Déniz, O., Castrillón, M. & Hernández, M. (2003). Face recognition using independent component analysis and support vector machines. Pattern Recognition Letters, 24(13), 2153–2157.

de Souza e Silva, E. G., Legey, L. F. L., & de Souza e Silva, E. A. (2010). Forecasting oil price trends using wavelets and hidden Markov models. Energy Economics, 32(6), 1507–1519.

Deutsch, M., Granger, C.W.J. & Teräsvirta, T. (1994). The combination of forecasts using changing weights. International Journal of Forecasting, 10(1), 47–57.

Duan, W-Q. & Stanley, H.E. (2011). Cross-correlation and the predictability of financial return series. Physica A: Statistical Mechanics and its Applications, 390(2), 290–296.

Enke, D. & Thawornwong, S. (2005). The use of data mining and neural networks for forecasting stock market returns. Expert Systems with Applications, 29(4), 927–940.

Eoma, C., Choi, S., Ohb, G., Jung, W.-S. (2008). Hurst exponent and prediction based on weak-form efficient market hypothesis of stock markets. Physica A, 387, 4630–4636.

Esfahanipour, A. & Aghamiri, W. (2010). Adapted neuro-fuzzy inference system on indirect approach TSK fuzzy rule base for stock market analysis. Expert Systems with Applications, 37, 4742–4748.

Fama, E. F. (1965) Portfolio analysis in a stable Paretian market, Management Science, 11, (3 Series A), 404–419.

Fama, E.F. (1970). Efficient capital markets: a review of theory and empirical work. The Journal of Finance, 25(2), 383–417.

Freitas, F.D., Souza, A. F. D., Almeida, A. R. D. (2009). Prediction-based portfolio optimalization model using neural networks. Neurocomputing, 72, 2155-2170.

Granger, C.W.J. & Ramanathan, R. (1984). Improved methods of combining forecasts. Journal of Forecasting, 3(2), 197–204.

Guo, Z., Zhao, W., Lu, H. & Wang, J. (2012). Multi-step forecasting for wind speed using a modified EMD-based artificial neural network model. Renewable Energy, 37(1), 241–249.

Hadavandi, E., Shavandi, H. & Ghanbari, A. (2010). Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting. Knowledge-Based Systems 23(8), 800–808.

Hansen, J. V., & Nelson, R. D. (2002). Data mining of time series using stacked generalizers. Neurocomputing, 43(1), 173–184.

Hall, J. W. (1994). Adaptive selection of US stocks with neural nets. Trading on the edge: Neural, genetic and fuzzy systems for chaotic financial markets,45–65.

Hagenau, M., Liebmann, M., Neumann, D. (2013). Automated news reading:Stock prie prediction based on financial news using context-capturing features. Decision Support Systems, 55(3), 685-697.

Halliday, R. (2004). Equity trend prediction with neural networks. Research Letters in the Information and Mathematical Sciences, 6.

He, K., Xie, C., Chen, S. & Lai, K. K. (2009). Estimating VaR in crude oil market: A novel multi-scale non linear ensemble approach incorporating wavelet analysis and neural network. Neurocomputing, 72(16-18), 3428–3438.

Hellstrom, T. (2000) Predicting a rank measure fo rstock returns,Theory of Stochastic Processes22(6)64–83.

Horváth, P. (2012). Tőzsdei portfolio kialakítása neurális hálózatok segítségével. Szakdolgozat.

Hsieh., T-J., Hsiao, H-F., & Yeh, W-C. (2011). Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm. Applied Soft Computing, 11(2), 2510–2525.

Hsu, S.-H., Hsieh, JJ. P.-A., Chih, T.V., Hsu, K.-C. (2009). A two-stage architecture for stock price forecasting by integrating self-organizing map and support vector regression. Expert Systems with Applications, 36, 7947-7951

Huang, N.E., Shen, Z., Long, S.R.,Wu, M.C., Shih, H.H., Zheng, Q., Yen, N.C., Tung, C.C. & Liu, H.H. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and nonstationary time series analysis. Proceedings of the Royal Society of London A - Mathematical Physical and Engineering Sciences, Series A, 454, 903–995.

Huang, W., Nakamori, Y. & Wang, S-Y. (2005). Forecasting stock market movement direction with support vector machine. Computer and Operations Research, 32, 2513–2522.

Huang, C-L., & Tsai, C-Y. (2009). A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting. Expert Systems with Applications, 36(2), 1529–1539.

Huang, S-C., Chuang, P-J., Wu, C.F. & Lai, H-J. (2010). Chaos-based support vector regressions for exchange rate forecasting. Expert Systems with Applications, 37(12), 8590–8598.

Huck, N. (2010). Pairs trading and outranking:The multi-step ahead forecasting case. European Journal of Operational Research, 207(3), 1702-1716.

Hyvarinen, A., Karhunen, J. & Oja, E. (2001). Independent component analysis, John Wiley and Sons, NewYork.

IBM, (2011). IBM SPSS Modeler CRISP-DM Guide.

Jaeger, H. (2002). A tutorial on training recurrent neural networks covering BPPT, RTRL, EKF and the "echo state network" approach. Technical Report, GMD Forschungszentrum Informationstechnic GmbH.

James, C. J. & Gibson, O. J. (2003). Temporally constrained ICA: An application to artifact rejection in electromagnetic brain signal analysis. IEEE Transactions on Biomedical Engineering, 50(9), 1108–1116.

J.M. Bates, C.W.J. Granger, (1969) The combination of forecasts. Operational Research Society, 20(4), 451–468.

Kanas, A. & Yannopoulos, A. (2001). Comparing linear and nonlinear forecasts for stock returns. International Review of Economics and Finance, 10(4), 383–398.

Kapelner, T. & Madarász, L. V. (2012). Független komponens analízis és empirikus tesztjei kötvényhozamok felhasználásával. TDK dolgozat.

Kazem, A., Sharifa, E., Hussain, F. K., Saberi, M. & Hussain, O. K. (2013). Support vector regression with chaos based firefly algorithm for stock marketprice forecasting. Aplied Soft Computing, 13(2), 947-958.

Khashei, M., Bijari, M. & Ardali, G.A.R. (2009). Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs). Neurocomputing, 72(4–6), 956–967.

Kim, K-J. & Han I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. Expert Systems with Applications, 19(2), 125–132.

Kim, K-J. (2003). Financial time series forecasting using support vector machines. Neurocomputing, 55(1–2), 307–319.

Koller, D. & Sahami, M. (1996). Toward optimal feature selection, in: Proceedings of the Thirteenth International Conference on Machine Learning (ML), Bari, Italy.

Kon, S. J. (1984) Modelsofstockreturns-a comparison,TheJournalofFinance, 39(1), 147–165.

Kondratenko, V. V. & Kuperin, Y. A. (2003), Using recurrent neural networks to forecasting of forex, Technical report, arXiv.org.

Kosaka, M., Mizuno, H., Sasaki, T., Someya, R. & Hamada, N. (1991). Applications of fuzzy logic/neural network to securities trading decision support system. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 1913–1918.

Min, J. H., & Lee, Y-C. (2005). Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. Expert Systems with Applications, 28(4), 603–614.

Mok, P.Y., Lam, K.P. & Ng, H.S. (2004). An ICA design of intraday stock prediction models with automatic variable selection. In: Proceedings of 2004 IEEE International Joint Conference on Neural Networks, Budapest, Hungary.

Moody, J. (1995). Economic forecasting: Challenges and neural network solutions. In Proceedings of the International Symposium on Artificial Neural Networks.

Moody, J., Saffell, M. (2001). Learning to trade via direct reinforcement, IEEE Transactions on Neural Networks 12 (4), 875–889.

Lee, M-C. (2009). Using support vector machine with a hybrid feature selection method to the stock trend prediction. Expert System with Applications, 36(8), 10896-10904.

Leigh, W., Purvis, R. & Ragusa, J.M. (2002). Forecasting the NYSE composite index with technical analysis, pattern recognizer, neural network, and genetic algorithm: a case study in romantic decision support. Decision Support Systems, 32(4), 361–377.

Li, T., Li, Q., Zhu, S. & Ogihara, M. (2003). A survey on wavelet applications in data mining. SIGKDD Explorations, 4(2), 49–68.

Lin, X., Yang, Z., Song, Y. (2009). Short-term stock price prediction based on echo state networks. Expert Systems with Applications, 36, 7313–7317.

Liu, Y. & Zheng, Y. F. (2006). FS_SFS: A novel feature selection method for support vector machines. Pattern Recognition, 39(7), 1333–1345.

Lu, C-J., Lee, T-S. & Chiu, C-C.(2009). Financial time series forecasting using independent component analysis and support vector regression. Decision Support Systems 47(2), 115–125

Lu, C-J. (2010). Integrating independent compontnt analysis-based denoising scheme with neural network for stock price prediciton. Expert Systems with Applications, 37(10), 7056-7064

Nebehaj, V. (2010). Pénzügyi és gazdasági idősorok előrejelzése neurális hálózatok segítségével. Szakdolgozat.

Ni, H. & Yin, H., (2009). Exchange rate prediction using hybrid neural networks and trading indicators. Neurocomputing, 72(13–15), 2815–2823.

Ni, L-P., Ni, Z-W. & Gao, Y-Z. (2011). Stock trend prediction based on fractal feature selection and support vector machine. Expert System with Applications, 38, 5569-5576

Oh, K. J. & Kim, K.-J. (2002). Analyzing stock market tick data using piecewise nonlinear model. Expert System with Applications, 22(3), 249–255.

Oja, E., Kiviluoto, K., & Malaroiu, S., (2000). Independent component analysis for financial time series. In Proceeding of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium, Lake Louise, Canada. 111–116.

Olson, D. & Mossman, C. (2003). Neural network forecasts of Canadian stock returns using accounting ratios. International Journal of Forecasting, 19(3), 453–466.

Pan, H. (2003). A joint review of technical and quantitative analysis of the financial markets towards a unified science of intelligent finance. Hawaii International Conference on Statistics and Related Fields, Hawaii, USA.

Panchal, G., Ganatra, A., Kosta, Y. & Panchal, D. (2010). Searching most efficient neural network architecture using Akaike's Information Criterion. International Journal of Computer Applications, 1(5), 8875-8887.

Pantazopoulos, K., Tsoukalas, L., Bourbakis N., Brun, M., Houstis,E. (1998). Financial prediction and trading strategie susing neuro fuzzy approaches,IEEE Transactions on Systems,Man and Cybernetics - Part B:Cybernetics28(4) 520–531.

Petróczi, A. I. (2009). Pénzügyi idősorok előrejelzése adatbányászati módszerekkel. Diplomaterv.

Phua, P., Hoh, K., Daohua, M. & Weiding, L. (2001). Neural network with genetically evolved algorithms for stocks prediction. Asia-Pacific Journal of Operational Research, 18(1).

Qin, Q., Wang, Q-G., Li, J. & Ge. S. S. (2013). Linear and Nonlinear Trading Models with Gradient Boosted Random Forests and Application to Singapore Stock Market. Journal of Intelligent Learning Systems and Applications, 5, 1-10.

Sermpinis, G., Dunis, C., Laws, J. & Stasinakis, C. (2012). Forecasting and trading the EUR/USD exchange rate with stochastic Neural Network combination and time-varying leverage. Decision Support Systems, 54(1), 316-329.

Sermpinis, G.,Theofilatos, G., Karathanasopoulos, A., Georgopoulos, E. F., Dunis C. (2013) Forecasting foreign exchange rates with adaptive neural networksusing radial-basis functions and Particle Swarm Optimization. European Journal of Operational Research, 225, 528-540.

Sharpe, W. F., Alexander, G. J., Bailey, V. (1999). Investments, sixth ed., Prentice-Hall, Upper Saddle River. NewJersey.

Sheu, H.-J., Wei, Y.-C. (2011). Effective options trading strategies based on volatility forecasting recruiting investor sentiment. Eypert Systems with Application, 38(1), 585-596.

Sitte, R. & Sitte, J. (2002). Neural networks approach to the random walk dilemma of financial time series. Applied Intelligence, 16(3), 163–171.

Swanson, N.R. & Zeng, T. (2001). Choosing among competing econometric forecasts: regression-based forecast combination using model selection. Journal of Forecasting, 20(6), 425–440.

Tan, Z.T., Quek, C. & Ng, G.S. (2007). Biological brain-inspired genetic complementary learning for stock market and bank failure prediction. Computational Intelligence, 23(2), 236–261.

Tay, F. E. H., & Cao, L. J. (2001). Improved financial time series forecasting by combining support vector machines with self-organizing feature map. Intelligent Data Analysis, 5, 339–354.

Timmermann A. (2006) Chapter 4: Forecast Combinations. Handbook of Economic Forecasting, 1, 135–196.

Thawornwong, S. & Enke, D. (2004) The adaptive selection of financial and economic variables for use with artificial neural networks. Neurocomputing, 56, 205–232.

Vapnik, V. N. (1995). The nature of statistical learning theory, Second ed., Springer-Verlag, New York

Vapnik, V., Golowich, S. & Smola, A., (1997). Support vector method for function approximation, regression estimation, and signal processing. In: Mozer, M., Vapnik, V. (Eds.), The Nature of Statistical Learning Theory, Second ed., Springer-Verlag, New York.

Varga, B. (2009). Tendenciák a neurális halo alapú kereskedési stratégiák profitábilitásában.

Varga, P. (2011). Tőzsdei idősorok előrejelzése adatbányászati módszerekkel. Szakdolgozat.

Vellido, A., Lisboa, P. J. G. & Vaughan, J. (1999). Neural networks in business: A survey of applications (1992–1998). Expert Systems with Applications, 17(1), 51–70.

Vincent, H.T., Hu, S-L.J. & Hou, Z. (1999). Damage detection using empirical mode decomposition method and a comparison with wavelet analysis. Proceedings of the Second International Workshop on Structural Health Monitoring, Stanford, 891–900.

Wang, Y-F. (2003). Mining stock prices using fuzzy rough set system. Expert System with Applications, 24(1), 13–23.

Wang , J-J., Wang, J-Z., Zhang, Z-G. & Guo, S-P. (2012). Stock index forecasting based on a hybrid model. Omega, 40(6), 758-766.

Wang, J-Z., Wang, J-J., Zhang, Z-G. & Guo, S-P. (2011). Forecasting stock indicies with bak propagation neural network. Expert Systems with Applications, 38(11), 14346-14355.

Wikowska, D. (1995). Neural networks as a forecasting instrument for the Polish Stock Exchange. International Advances in Economic Research, 1(3), 232–242.

Yaser, S. A-M. & Atiya, A. F. (1996). Introduction to financial forecasting. Applied Intelligence, 6, 205–213.

Yousefi, S., Weinreich, I. & Reinarz, D. (2005). Wavelet-based prediction of oil prices. Chaos, Solitions and Fractals, 25(2), 265-275.

Yu, L., Wang, S.Y., Lai, K.K., (2005). A novel nonlinear ensemble forecasting model incorporating GLAR and ANN for foreign exchange rates. Computers & Operations Research, 32(10), 2523–2541.

Yu, L., Wang, S. & Lai, K.K. (2008). Forecasting crude oil price with an EMD-based neural network ensemble learning paradigm. Energy Economics, 30(5), 2623-2635.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, 14, 35–62.

Zhang, Y. D., & Wu, L. N. (2009). Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network. Expert Systems with Applications, 36, 8849–885

9. Mellékeltek, táblázatok

	Max	Min	Átlag	Szórás
Súlyozott MA	20460	11805	16581	1710
Momentum	2480	-1905	-22,4	769,1
Stochastic K%	100	0	46,7	30
Stochastic D%	97,6	3,5	46,7	25,6
RSI	87,9	2,2	48,1	15
MACD	646,5	-658,7	-36,9	247,0
LW R%	0	-100	-54	29,6
A/D Oszcillator	100	0	48,2	28,4

15.táblázat A technikai indikátorok statisztikai jellemzői (Saját szerkesztés)

Indikátor neve	Formula
Súlyozott 10 napos mozgóátlag	$\frac{((n) \times C_t + (n-1) \times C_{t-1} + \cdots + C_{10})}{(n + (n-1) + \cdots + 1)}$
Momentum	$C_{t-}C_{t-n}$
Sztochasztikus K%	$\frac{C_t - LL_{t-n}}{HH_{t-n} - LL_{t-n}} \times 100$
Sztochasztikus D%	$\frac{\sum_{i=0}^{n-1} K_{t-i}\%}{n}$
RSI	$100 - \frac{100}{1 + (\sum_{i=0}^{n-1} Up_{t-i}/n) \,/ (\sum_{i=0}^{n-1} Dw_{t-i}/n)}$
MACD	$MACD(n)_{t-1} + 2/n + 1 \times (DIFF_t - MACD(n)_{t-1})$
Larry William's R%	$\frac{H_n - C_t}{H_n - L_n} \times 100$
A/D oszcillátor	$\frac{H_t - C_{t-1}}{H_t - L_t}$

16.táblázat Az inputváltozók számítási módszere (Saját szerkesztés)

Metrika	Formula
RMSE	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(T_i - P_i)^2}{n}}$
MAPE	$MAPE = \frac{\sum_{i=1}^{N} \left\lvert \frac{T_i - P_i}{T_i} \right\rvert}{N}$
DA	$\frac{100}{n}\sum_{i=1}^{n} d_i, \quad \text{ahol } \mathrm{d_i} = \begin{cases} 1, & \text{ha } (\mathrm{P_i} - \mathrm{P_{i-1}})(\mathrm{T_i} - \mathrm{T_{i-1}}) \geq 0 \\ 0 & \text{egyébként} \end{cases}$

17.táblázat A kiértékelési mutatók számítási módszere (Saját szerkesztés)

Teljesítménymutató	Formula
Évesített hozam	$R^A = 252 \times \frac{1}{N} \times \left(\sum_{t=1}^{N} R_t\right)$, ahol R_t napi hozam
Évesített szórás	$\sigma^A = \sqrt{252} \times \sqrt{\frac{1}{N-1} \times \left(\sum_{t=1}^{N} (R_t - R')^2\right)}$

18.táblázat A kiértékelési mutatók számítási módszere (Saját szerkesztés)

Rejtett rétegben lévő neuronok száma	Tanulási ráta	Validációs RMSE
11	0,01	0,091147
	0,02	0,089377
	0,03	0,088919
	0,04	0,088731
	0,05	0,088628
12	0,01	0,092775
	0,02	0,091617
	0,03	0,091053
	0,04	0,090677
	0,05	0,090398
13	0,01	0,091326
	0,02	0,090223
	0,03	0,089775
	0,04	0,089509
	0,05	0,089324
14	0,01	0,091597
	0,02	0,089812
	0,03	0,088913
	0,04	0,088398
	0,05	**0,088076**

19.táblázat Különböző paraméterű BPN hálózatok hibája a teszthalmazon (MOL) (Saját szerkesztés)

Főkomponensek	MOL
IC1, IC2, IC3, IC4, IC5, IC6, IC7	**3,5241**
IC1, IC2, IC3, IC4, IC5, IC6, IC8	4,2944
IC1, IC2, IC3, IC4, IC5, IC7, IC8	5,7812
IC1, IC2, IC3, IC4, IC6, IC7, IC8	4,4403
IC1, IC2, IC3, IC5, IC6, IC7, IC8	6,9534
IC1, IC2, IC4, IC5, IC6, IC7, IC8	6,6435
IC1, IC3, IC4, IC5, IC6, IC7, IC8	7,551
IC2, IC3, IC4, IC5, IC6, IC7, IC8	3,6552

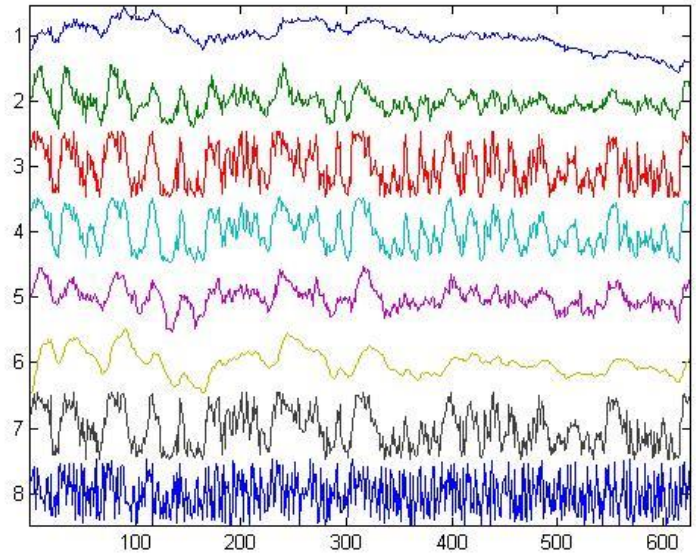
20.táblázat Különböző rekonstruált inputmátrixok RHD értékei (MOL) (Saját szerkesztés)

Rejtett rétegben lévő neuronok száma	Tanulási ráta	Validációs RMSE
11	0,01	0,092078
	0,02	0,089974
	0,03	0,089373
	0,04	0,089131
	0,05	0,089003
12	0,01	0,094122
	0,02	0,092606
	0,03	0,091745
	0,04	0,091206
	0,05	0,090835
13	0,01	0,091228
	0,02	0,090416
	0,03	0,090089
	0,04	0,089877
	0,05	0,089718
14	0,01	0,092137
	0,02	0,090300
	0,03	0,089358
	0,04	0,088831
	0,05	**0,088516**

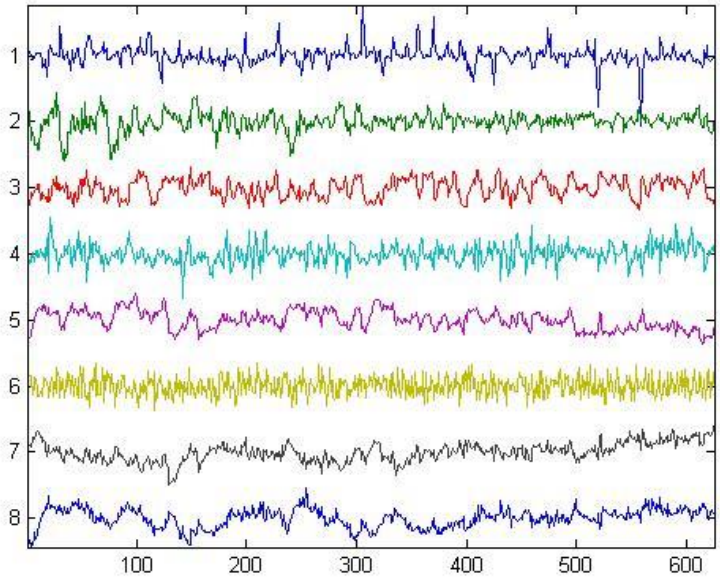
21.táblázat Különböző paraméterű ICA-BPN hálózatok hibája a teszthalmazon (MOL) (Saját szerkesztés)

IMF	Neuronok száma	Tanulási ráta
1	13	0,025
2	13	0,05
3	13	0,05
4	13	0,05
5	11	0,05
6	11	0,05
7	11	0,05

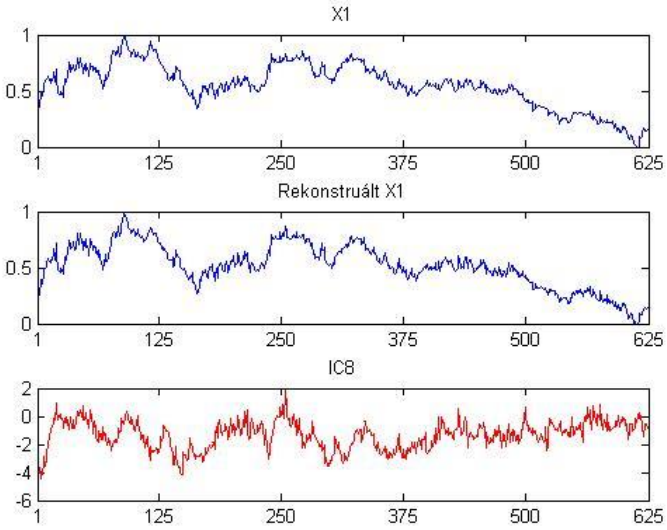
22.táblázat Különböző IMF-ek optimális paraméterei (MOL) (Saját szerkesztés)



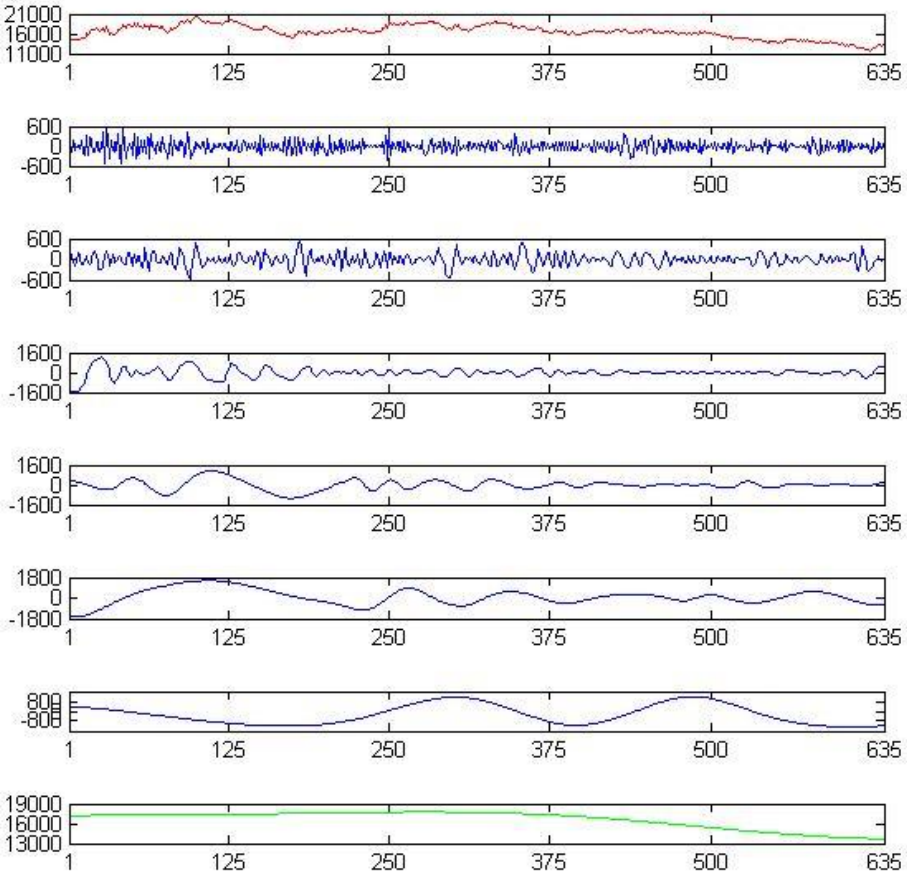
18.ábra A MOL modellezéséhez használt inputváltozók (Saját szerkesztés)



19.ábra Az inputváltozókból képzett független-komponensek (MOL) (Saját szerkesztés)



20.ábra Az eredeti és a rekonstruált x_1 változó illetve a zaj komponens (Saját szerkesztés)



21.ábra A MOL árfolyamainakempirikus alapú dekompoziciója (Saját szerkesztés)