

Univerzitet u Beogradu
Matematički fakultet

***Implementacija finansijskih strategija
za berzanska poslovanja***

MASTER RAD

Mentor: dr Vladimir Filipović

Student: Mrđan Stojimirović
br. indeksa: 1052/2011

Septembar 2013.

Apstrakt

Poslovanje na finansijskim tržištima predstavlja jedno od potencijalno najprofitabilnijih poslovanja na svetu. Bilo da se radi o hartijama od vrednosti, plemenitim metalima, poljoprivrednim proizvodima itd. finansijska tržišta omogućuju učesnicima da ostvare velike dobitke. Moglo bi se reći da je taj ogroman dobitni potencijal generisan ogromnim rizikom koji prihvataju učesnici na pomenutim tržištima, odnosno koji se bave berzanskim poslovanjem. Svako od njih bi voleo da može da pogleda u budućnost kako bi znao koje korake da preduzme da bi maksimizovao profit, odnosno minimizovao gubitke. Kako to nije moguće, učesnici na berzi se zadovoljavaju primenom tehničke i fundamentalne analize na posmatrano tržište u cilju što bolje prognoze sledećih dešavanja.

U ovom radu je predstavljen jedan od načina tehničke analize finansijskih tržišta kroz upotrebu matematičkih modela (ARIMA modela) u cilju dobijanja prognoze za sledeću očekivanu vrednost. Rad opisuje razvoj softverskog sistema za prognoziranje očekivane vrednosti, njegovu strukturu, kao i karakteristike razvijenog sistema.

Glavni cilj ovog rada je opis implementacije finansijskih strategija, čime se daje doprinos preciznijoj, dubljoj analizi tržišta koja bi dovela do lakšeg izvršavanja transakcija.

Ključne reči: finansijska tržišta, ARIMA modeli, softver, implementacija strategija

Abstract

Business on financial markets is one of the potentially most profitable business in the world. Whether it is about the securities, precious metals, agricultural products, etc. Financial markets allow participants to achieve big gains. One could say that the huge winning potential is generated by enormous risk accepted by participants in these markets, who are dealing with stock business. Each of them would love to be able to look into the future to know what steps to take to maximize profits or minimize losses. As this is not possible, market participants are satisfied with the application of technical and fundamental analysis of the observed market in order to better forecast the next event.

This paper presents a way of technical analysis of financial markets through the use of mathematical models (ARIMA) in order to obtain forecasts for the next expected value. This paper describes the development of a software system for predicting the expected value, the structure and characteristics of the developed system.

The main objective of this paper is a description of the implementation of financial strategies, thus contributing to more accurate, deeper analysis of the market that would lead to an improvement of transaction execution.

Keywords: financial markets, ARIMA models, software, implementation strategy

Sadržaj

Uvod.....	4
1. Finansijsko tržište	4
1.1. Dimenzije tržišta	5
1.2. Podela tržišta	5
1.3. FOREX tržište (Međunarodno tržište valuta)	6
2. MetaTrader 4.....	10
3. Formulacija problema	14
3.1. Funkcionalni zahtevi aplikacije ImFiStBer.....	14
4. Grafički interfejs aplikacije ImFiStBer.....	15
5. Implementacija strategija	23
5.1. Analiza grafikona koeficijenata autokorelacije i parcijalne autokorelacije. Izbor p,d,q vrednosti. 23	
5.2. Dicky-Fooler Test. Ispitivanje stacionarnosti AR(1) vremenske serije.	27
5.3. Modeli	31
5.3.1. AR(1)	31
5.3.2. AR(2)	32
5.3.3. MA(1)	34
5.3.4. ARMA(1,1) = ARIMA(1,0,1).....	35
5.3.5. ARIMA(1,1,1) i ARIMA(1,2,1).....	37
5.3.6. ARIMA(1,1,0) i ARIMA(1,2,0).....	39
5.3.7. ARIMA(0,1,1) i ARIMA(0,2,1).....	41
5.3.8. ARIMA(2,1,0) i ARIMA(2,2,0).....	43
6. Primer analize tržišta. Otvaranje i zatvaranje transakcije.	46
Zaključak	54
Literatura.....	55

Uvod

Da bi se moglo pristupiti analizi finansijskih tržišta, potrebno ih je prvo definisati, odrediti dimenzije i izvršiti odgovarajuću podelu. Tržište na koje će se ovaj rad fokusirati je FOREX tržište (Međunarodno tržište valuta). Jedna od globalno prihvaćenih aplikacija koje omogućavaju pristup i poslovanje na FOREX-u je MetaTrader 4. Ova aplikacija će biti korišćena za dobijanje podataka koji će kasnije biti modelirani i nad kojima će se isprobavati implementirane finansijske strategije.

U prvom poglavlju se opisuje finansijsko tržište, analiziraju se njegove dimenzije i vrši se podela sa geografskog aspekta i aspekta tržišnih struktura. Detaljno se opisuje i analizira poslovanje na FOREX (eng. *FOREign EXchange market*) tržištu.

Drugo poglavlje opisuje izgled aplikacije MetaTrader 4. Tu se opisuje i način kako sačuvati bazu podataka koja će se kasnije koristiti za modeliranje.

U trećem poglavlju se formuliše problem poslovanja na finansijskim tržištima i nude se matematički modeli za rešavanje problema kao i svi zahtevi koje softversko rešenje treba da zadovoljava.

Četvrto poglavlje sadrži opis grafičkog interfejsa aplikacije koja je razvijena za potrebe ovog master rada.

U petom poglavlju je opisan način implementacije strategija koje su navedene u trećem poglavlju.

Šesto poglavlje kroz primer opisuje otvaranje i zatvaranje transakcije koristeći aplikaciju koja je razvijena za potrebe ovog master rada.

1. Finansijsko tržište

Tržište se može definisati na različite načine. Različitost gledišta i definicija o tržištu izraz je poznate činjenice da je tržište kao fenomen veoma kompleksno, da postoje mnogobrojni aspekti analize tržišta i različite forme u kojima se tržište ispoljava.

Posmatrano sa aspekta odnosa koji se uspostavljaju na tržištu, definiše se kao:

”sveukupnost odnosa ponude i tražnje koji na određenom mestu i određenom vremenu utiču na prodaju pojedinih roba i skup svih ustanova područja i uređaja koji omogućuju organizovani i stalni kontakt između kupaca i prodavaca.”

Opšta teorija sistema opisuje tržište kao

“jedan složen dinamički i stohastički sistem, koji je istovremeno podsistem sistema funkcionisanja velikog ekonomskog sistema”.

Dati jednu potpunu definiciju tržišta značilo bi da se u definisanju, pored uzimanja u obzir njegovih komponenti, moraju uzeti u obzir i njegova struktura i funkcije.

1.1. Dimenzije tržišta

Brojne definicije tržišta ukazuju na njegove osnovne dimenzije. Osnovne dimenzije tržišta bez kojih se ono ne može definisati su:

- ljudi,
- platežna sposobnost,
- spremnost ljudi,
- vreme i
- prostor.

Ljudi se na tržištu mogu pojaviti kao kupci ili kao prodavci. Bez ljudi tržište ne bi moglo funkcionisati. Kako računari i savremena sredstva za komunikaciju sve više zauzimaju mesto čoveku može se očekivati da će tržište u budućnosti sve manje zavisiti od ljudi, a sve više od sredstava i informacija. Prema tome, za očekivati je da u budućnosti ova dimenzija – ljudi – bude delimično ili u potpunosti zamenjena novom dimenzijom – sredstva i informacije.

Platežna sposobnost daje ritam tržištu. U situacijama kada platežna sposobnost pada, opada i ritam kupovina i prodaja. Ovo ipak ne mora biti u direktnoj zavisnosti jer se ritam može regulisati i drugim instrumentima.

Spremnost ljudi da kupuju ili prodaju, takođe definiše tržište. Ljudi mogu imati novac (kupci) i robu (prodavci) – a da nisu spremni prodavati ili kupovati. Spremnost ljudi na veću kupovinu, je u direktnoj srazmeri sa kamatnom stopom – ako novac može da se dobije sa niskim kamatama, to će stimulisati potrošnju i obrnuto, visoke kamatne stope stimulisace štednju. Takođe i visoka inflacija se može spomenuti kao faktor koji stimuliše potrošnju.

Vreme je takođe bitna dimenzija tržišta. Imati kvalitetne proizvode i usluge, ali u pogrešno vreme, je isto kao i nemati ih uopšte ili još gore od toga. Imati proizvode i usluge u pogrešno vreme je čist promašaj jer smo sredstva i resurse nenamenski potrošili. Zbog toga moramo raspolagati sa tržišnim informacijama kako bismo pravu robu obezbedili u pravo vreme.

Prostor kao dimenzija tržišta je slična predhodnoj. Imati proizvode i usluge na pogrešnom mestu je čist promašaj. Bolje da ih upošte nismo imali, jer smo bespotrebno i pogrešno utrošili resurse.

Na osnovu predhodno navedenih dimenzija, tržište možemo definisati kao težnju preduzeća ili pojedinca da svoje proizvode i usluge plasira tamo gde su kupci koji su platežno sposobni i spremni za kupovinu na mestu koji oni žele i u vremenu kada im odgovara.

1.2. Podela tržišta

Zavisno sa kojeg stanovišta se vrši podela, tržišta se mogu klasifikovati na sledeći način:

1. **Posmatrano sa geografskog aspekta**, tržišta se dele na:

- **lokalno tržište**, koje karakteriše ponuda i tražnja na užem lokalnom području (grad, opština).
- **regionalno tržište** predstavlja skup lokalnih tržišta, pri čemu se ponuda i tražnja odvijaju na regionalnom prostoru.
- **nacionalno tržište**, unutrašnje ili domaće, predstavlja skup regionalnih tržišta u okviru jedne zemlje, pri čemu domaća ponuda ne mora biti usmerena na domaću tražnju, već može biti usmerena i na

svetsku tražnju. Domaća tražnja se takođe može alimentirati iz inostranih izvora, a ne samo iz domaće ponude.

- **svetsko tržište** predstavlja tržište svih zemalja sveta, pri čemu svetsku ponudu, odnosno tražnju, sačinjavaju ukupna ponuda, odnosno tražnja proizvoda ili usluga svih zemalja sveta.
2. **Sa aspekta tržišnih struktura** može se izvesti podela na perfektno (tržište neograničene, potpune konkurencije) i imperfektno tržište (tržište ograničene konkurencije).

Perfektno tržište karakterise razmena po tržišnim cenama koje se obrazuju kao sinteza postojećih odnosa ponude i tražnje. Zbog velikog broja učesnika, kako na strani ponude, tako i na strani tražnje, niko od njih svojim individualnim postupcima ne može uticati na promenu tržišnih cena. Za mnoge ekonomiste XIX veka tržište perfektne konkurencije je bilo ideal. Međutim, stvarno stanje na tržištu znatno odstupa od takvog tipa tržišta i predstavlja smesu konkurencije, monopola i oligopola.

Osnovne primese postojanja tržišta perfektne konkurencije su:

- autonomnost ponašanja tržišnih subjekata - što podrazumeva punu slobodu tržišnih učesnika, odnosno odsustvo administrativnih i restriktivnih mera kojima se sužava autonomija ponašanja privrednih subjekata;
- ravnoteža ponude i tražnje – ukupna ponuda treba da je jednaka ukupnoj tražnji;
- supstitivnost – podrazumeva potpunu indentičnost jednog proizvoda kod svih proizvođača koji ga proizvode, što ima za posledicu da je presudan kriterijum njihovog izbora prodajna cena;
- slobodna cirkulacija kapitala – predstavlja odsustvo bilo kakvih prepreka seljenju kapitala iz jedne privredne grane u drugu, iz jednog preduzeća u drugo;
- atomiziranost ponude i tražnje – podrazumeva brojnost i prodavaca i kupaca, tako da svako od njih, u odnosu na druge, ne može biti u poziciji da utiče na povećanje ili smanjenje kupovine, odnosno prodaje, na nivo i odnose tržišnih cena;
- fleksibilnost cena – znači da se u zavisnosti od rasta ili pada ukupne ponude i tražnje, menjaju i cene;
- transparentnost tržišta – predpostavlja da su svim tržišnim učesnicima poznati i dostupni svi faktori relevantni za formiranje ponude i tražnje;
- selektivnost tržišta – podrazumeva mogućnost eliminacije pojedinih tržišnih učesnika, koji nisu u stanju da opstanu na tržištu.

Imperfektno tržište karakterišu: ograničenost konkurencije, što dovodi do snažnijeg uticaja ponuđača, odnosno kupaca na proces formiranja tržišnih cena, diferenciranje proizvoda i cena iste namene, tako da se isti proizvod i usluge iz potpuno indentičnih uslova i koji bi trebalo da se realizuju po jednakim prodajnim cenama, prodaju i kupuju po različitim cenama i relativno slabija međugranska obilnost sredstava.

1.3. FOREX tržište (Međunarodno tržište valuta)

FOREX (eng. *FOREign EXchange market*) predstavlja globalno, decentralizovano finansijsko tržište valuta [8]. Može se reći da je FOREX tržište novca. Svaka valuta se vezuje sa drugom valutom i čini jedan valutni par. Postoje dve podele valutnih parova. Prva podela je na velike i ostale. Veliki valutni parovi su oni u kojima figurise jedna od sledećih valuta (EUR-evro, USD-američki dolar, CHF-švajcarski franak, AUD-australijski dolar, GBP-britanska funta, JPY-japanski jen i CAD-kanadski dolar). Primer: EURUSD, EURJPY, AUDCAD, GBPCHF su veliki parovi dok bi NZDSEK bio primer za ostale (NZD-novozelandski dolar, SEK-švedska kruna). Druga podela bi bila na parove u kojima figurise američki dolar i ostale. Primer: EURUSD, USDJPY, GBPUSD gde imamo USD i recimo EURJPY, EURGBP

nema USD.

FOREX je najveće finansijsko tržište na svetu i to u svakom pogledu. Primera radi, aprila 2010 dnevni prosek je iznosio oko \$3.98 trilliona dok je na NYSE (*New York Stock Exchange – berza akcija*) bio \$400 milijardi. Razlog za to je što, dok su druga tržišta ugašena usled kraja radnog vremena berzi, FOREX je otvoren 24h, 5 dana u nedelji. To se postiglo organizacijom rada u četiri vremenske zone koje se međusobno preklapaju. Zone nose imena po velikim gradovima koji se smatraju kao centri FOREX tržišta. To su Sidnej, Tokio, London i Njujork. Tržište se otvara u Sidneju, nedeljom u 22:00h, a zatvara se u Njujorku, petkom u 22:00h. Prvo se otvara Sidnej koji traje od 22:00h do 06:00h, paralelno traje i Tokio od 00:00h do 08:00h, zatim se otvara London od 08:00h do 18:00h koji se preklapa sa Njujorkom koji traje od 14:00h do 22:00h i onda se opet otvara Sidnej. Iako postoje 4 zone, najveći obrt kapitala je za vreme istovremenog trajanja Evropske i Američke zone odnosno od 14:00h do 18:00h. Situacija je takva zato što većina država poredi svoje valute prema dolaru odnosno evru. Zbog ovakve organizacije, učesnici širom sveta prouzrokuju promenu cena koja dovode do pada ili rasta potražnje jedne valute prema drugoj.

Ko su učesnici na FOREX tržištu? Svakako da su najveći učesnici same države, odnosno centralne banke koje u ime države učestvuju na valutnom tržištu. One prodajom ili kupovinom domaće ili strane valute direktno utiču na razvoj svoje ekonomije. Primer toga bi bila Kina sa njenom valutom juan. Mnogi optužuju Kinu da zbog obaranja vrednosti svoje valute utiče na konkurentnost svojih proizvoda. Na ovaj način Kina veštački “proizvodi” jeftinu radnu snagu. Obe ove činjenice dovode do velike potražnje kineskih proizvoda i velikih ulaganja u kinesku ekonomiju. Mnoge multinacionalne kompanije svoje proizvodne linije premeštaju u Kinu gde sa malom količinom svoje domaće valute, npr EUR, mogu da isplate više radnika nego što bi to mogli u svojoj zemlji. Drugi veliki učesnici na FOREX tržištu su hedž fondovi koji sa svojim ogromnim kapitalom prouzrokuju promene na tržištu. Velike multinacionalne kompanije takođe imaju veliki uticaj na berzu. Ako se posmatra poslovanje neke poznate kompanije kao što su Google ili Microsoft ili možda Ford, Fiat ili Audi, svaka od njih ima interesa da po najpovoljnijem kursu izvrši zamenu jedne valute u drugu kako bi ostvarili maksimalan profit. Na primer, ako kompanija Toyota izveze svoje proizvode na američko tržište, njoj je u interesu da svoj profit koji je u ovom slučaju u dolarima (USD) zameni po najpovoljnijim uslovima za jene (JPY). Male promene u kursu dovode do velikih gubitaka ili dobitaka kada se radi o velikoj količini novca. Primer je sledeći, dana 29.12.2010. odnos između USD i JPY bio je 82.348 dok je 30.12.2010 odnos između ove dve valute bio 81.613 . Ako bi u ovom slučaju Toyota izvršila konverziju od npr. 1.000.000,00\$ u japanske jene, ona bi 29.12. dobila 82.348.000,00¥, a da istu transakciju obavi 30.12. dobila bi 81.613.000,00¥ što bi značilo da je u roku od 24h kompanija mogla da načini dobitak ili gubitak od 735.000,00¥ u zavisnosti od toga kada su rešili da konvertuju USD u JPY. Još jedna grupa učesnika čine brokeri koji omogućavaju običnim ljudima da učestvuju na FOREX tržištu.

Neki osnovni pojmovi koji su potrebni za razumevanje tržišta su bikovi (eng. Bulls), medvedi (eng. Bears), prodajni kurs (eng. Bid), kupovni kurs (eng. Ask), razlika kupovnog i prodajnog kursa (eng. Spread), standardna količina novca potrebna za otvaranje transakcije (eng. Lot), odnos između uložених sredstava i sredstava sa kojima raspolažemo (eng. Lverage).

FOREX tržište se razlikuje od ostalih po tome što npr. kada cene akcija neke kompanije rastu, svi akcionari zarađuju i obrnuto kada cene padaju svi gube, dok na Forexu da bi jedan zaradio drugi mora da izgubi [9]. Primera radi, ako je Toyota ostvarila zaradu od 735.000,00¥ , neko drugi je morao da izgubi istu količinu novca. Možda je Ford izvezao svoje automobile na japansko tržište, pa je po nepovoljnom kursu izvršio konverziju valuta i ostvario je gubitak od 735.000,00¥ odnosno 8925,53\$. Ta neprestana “bitka” na tržištu, gde jedni dobijaju dok drugi gube, se vodi između Bikova (eng. Bulls) i Medveda (eng. Bears). Ljudi koji kupuju prvu valutu u nekom valutnom paru (npr EUR u paru EURUSD) nazivaju se

bikovi. Kako je tražnja za prvom valutom velika (ljudi je kupuju), to dovodi do rasta cena, odnosno do jačanja te valute prema drugoj. Tada grafikon koji predstavlja valutni par raste i mnoge to asocira na bikove koji jurišaju na gore. Ljudi koji prodaju prvu valutu u nekom valutnom paru (npr EUR u paru EURUSD) nazivaju se medvedi. Kako je tražnja za prvom valutom mala (ljudi je prodaju) to dovodi do pada cena odnosno do slabljenja te valute prema drugoj. Tada grafikon koji predstavlja valutni par pada i mnoge to asocira na medvede koji jurišaju na dole.

Za svaki valutni par se vezuju dve cene, koje predstavljaju kupovni i prodajni kurs. Primer za EURUSD 1.3254 / 1.3257. Poslednja decimala koja ima ulogu u formiranju cene naziva se Pip (eng. Percentage In Point) i to je u ovom slučaju četvrta decimala odnosno 1Pip je 0.0001. Prva cena 1.3254 predstavlja prodajni kurs (eng. Bid). To znači da kada bismo prodavali 1€ dobili bismo 1.3254\$. Druga cena 1.3257 predstavlja kupovni kurs (eng. Ask). To znači da nam za 1€ treba 1.3257\$. Razlika između kupovnog i prodajnog kursa (eng. Spread) je u ovom slučaju 0.0003 odnosno 3 Pip-a. Razliku između kupovnog i prodajnog kursa (eng. Spread) je ustvari zarada koju ostvaruje broker koji nam je omogućio učešće na valutnom tržištu. Brokери širom sveta pokušavaju smanjivanjem razlike između kupovnog i prodajnog kursa da postanu konkurentniji i da na taj način privuku nove klijente. Standardna količinu novca u jednoj transakciji (eng. Lot) uvek iznosi 100.000,00 prve valute nekog valutnog para, odnosno ako menjamo EUR u USD, znači da menjamo 100.000,00 € za određenu količinu dolara. Usluga koju nam pruža broker kao odnos između uložених sredstava i sredstava sa kojima raspolažemo za jednu transakciju (eng. Leverage) mogu biti 1:10, 1:50, 1:100, 1:200, 1:500. Primera radi, ako nam je broker omogućio odnos između uložених sredstava i sredstava sa kojima raspolažemo 1:500, to znači da se standardna transakcija za EURUSD može ostvariti sa svega 200EUR na računu. To otvara mogućnost da se sa malim investicijama ostvare veliki prihodi. Jedna stvar koja je veoma važna da se shvati je ta da broker nikada ne gubi. On ima sigurnu zaradu od nekoliko Pip-a (eng. Percentage In Point) koliko iznosi razlika između kupovnog i prodajnog kursa za određen valutni par i to je sva zarada koju on ostvaruje. Recimo ako je na računu stanje od 200€ i otvorena je neka transakcija koja je neprofitabilna, broker dozvoljava transakciji da gubi u vrednosti do 200€. Onog trenutka kada se transakcija približi toj cifri broker automatski gasi transakciju, račun je pretrpeo gubitak dok je kod nekog drugog npr. kompaniji Google račun u dobitku.

Način ostvarivanja zarade je sledeći. Kada grafikon raste ide se na kupovinu prve valute, a kad grafikon pada ide se na prodaju prve valute [10]. Uzmimo npr. EURUSD po ceni 1.0000/1.0003. Ako se po ovim uslovima izvrši kupovina 1€, za njega se potroši 1.0003\$. Nakon nekog vremena ako dođe do rasta cena i one iznose npr 1.5000/1.5003 i u tom trenutku se izvrši prodaja tog jednog evra, za njega se dobija 1.5000\$, znači ostvarena je zarada od 0.4997\$. Recimo da je od tog trenutka evro krenuo da slabi. Tada se izvrši prodaja 1€ i za njega se dobije 1.5000\$. Nakon nekog vremena kada se cene opet vrate na 1.0000/1.0003 izvrši se kupovina 1€ za koji se potroši 1.0003\$ nakon čega račun raspolaže sa početnim 1€ i 0.4997\$ profita. U stvarnom svetu profit se ostvaruje tako što se kupuju i prodaju milionske sume (10 i više Lot-ova), a zarada se ostvaruje na promeni cena u poslednje dve decimale jer za ovakve promene u ceni koje su date u primeru potrebno je mnogo vremena koje se meri u godinama.

Postoje dva pristupa analizi FOREX tržišta u cilju prognoze budućih dešavanja, To su fundamentalna i tehnička analiza.

Fundamentalna analiza je uticaj objavljenih ekonomskih vesti na tržište, dok se tehnička analiza oslanja na neke matematičke zakone. Dok za svako tržište postoji specifična fundamentalna analiza, tehnička analiza se može primeniti na sva tržišta za koja imamo dostupnu istoriju, tj. cene u prošlosti. Neke od vesti koje se objavljuju i koje utiču na FOREX tržište su:

- Bruto nacionalni dohodak (eng. GDP)
- Broj nezaposlenih (eng. Unemployment Claims)
- Prodaja novoizgrađenih kuća (eng. New Home Sales)

Pod tehničkom analizom se podrazumeva:

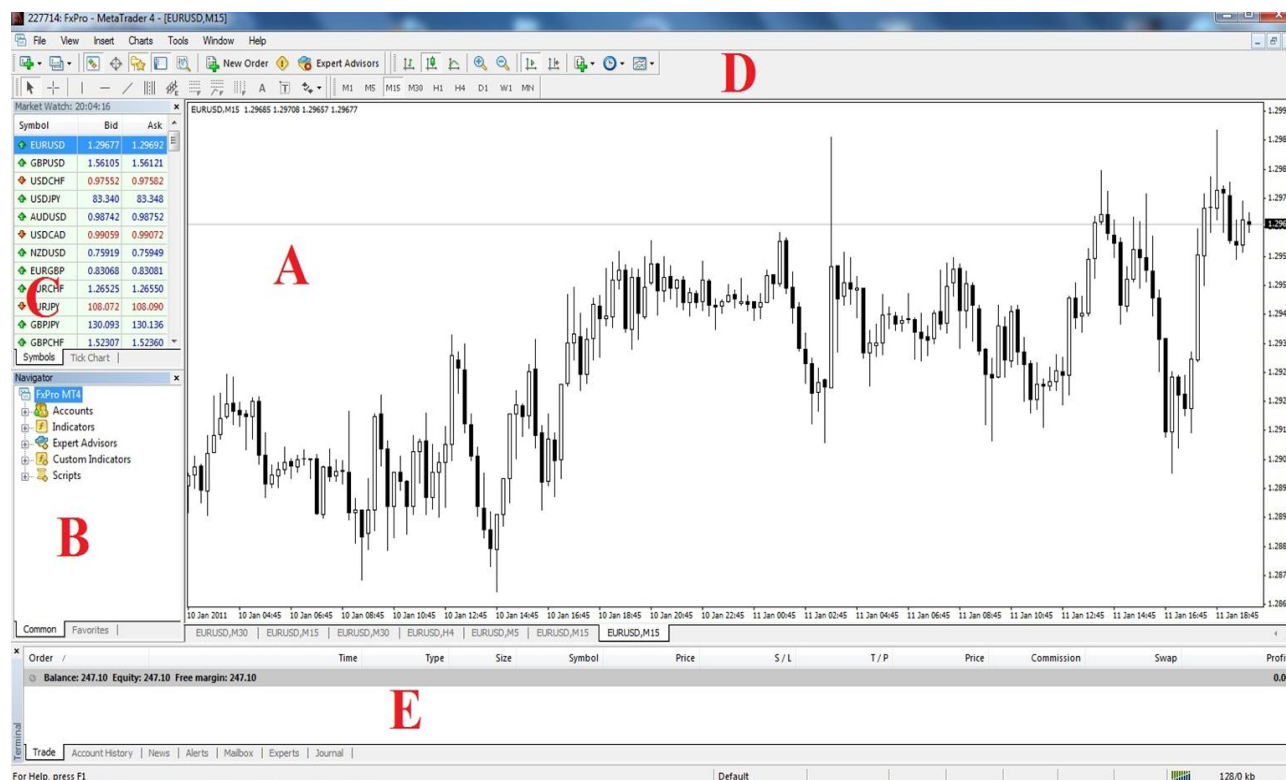
- konstrukcija grafikona
- analiza svećica
- crtanje linije otpora i linije podrške
- crtanje trend linija
- crtanje tunela
- analiza vraćanja i produžetaka (Fibonači)
- analiza šablona
- talasna teorija Eliota

i još mnogo toga.

2. MetaTrader 4

MetaTrader 4, ili **MT4**, je softverski program za elektronsku trgovinu na finansijskim tržištima. Moguće je trgovati sa plemenitim metalima, valutnim parovima, akcijama, hartijama od vrednosti, naftom itd. Razvila ga je softverska kompanija MetaQuotes Software. Prethodna verzija koju su izdali bila je MetaTrader 3 i ona više nije u upotrebi. Verzija koja se razvija sa ciljem da zameni postojeću je MetaTrader 5 koja je još uvek u fazi testiranja.

Softversko rešenje uključuje editor i kompajler za programski jezik MQL 4 pomoću kojeg se pišu indikatori, skripte, programi koji sami otvaraju i zatvaraju transakcije i na taj način olakšavaju rad.



Slika 1. Izgled programa

Najveći deo prozora aplikacije zauzima region sa grafikonom (A).

Region sa navigatorom (B) sadrži veći broj elemenata:

1. Accounts – oдавde korisnik bira sa kog računa posluje, jer platforma podržava mogućnost da se sa više računa posluje u isto vreme
2. Indicators – ovde mogu da se biraju indikatori koji su isporučeni zajedno sa platformom
3. Expert Advisors – ovde se nalaze programi (finansijski roboti) koji sami mogu da otvaraju i zatvaraju transakcije u ime korisnika
4. Custom Indicators – ovde se nalaze indikatori koje korisnik sam napiše ili dovuče sa Interneta
5. Scripts – ovde se nalaze programi specijalne namene npr. gašenje svih otvorenih transakcija jednim “klikom”

Region (C) predstavlja tabelu iz koje se posmatra valutni par (eng. Symbol), kupovni i prodajni kurs.

Odavde se otvaraju transakcije. Dvostrukim klikom na odgovarajući valutni par otvara se nov prozor u kome se bira da li se kupuje ili se prodaje izabrani valutni par, nakon čega se otvara transakcija.

U region označenim sa (D) se nalaze osnovni alati za tehničku analizu. Neki od njih su: crtanje horizontalnih, vertikalnih, trend linija, biranje odgovarajućeg grafikona, biranje vremenskog intervala za grafikon i fibonačijev alat. Ovi alati se mogu prilagoditi korisnikovim potrebama.

Region (E) ima više opcija od kojih su najvažnije Trade i Account History. U Trade opciji se vide sve otvorene transakcije i trenutna zarada ili gubitak u otvorenim transakcijama dok u Account History mogu da se vide sve transakcije od trenutka otvaranja računa i odatle može da se prati mesečna zarada (pozitivna ili negativna).

Indikatori koji dolaze uz platformu se dele na one koje prate trend i one koji prate oscilacije u cenama: Neki od indikatora su: Bollinger Bands, Moving Average, MACD, Stochastic Oscillator, RSI (Relative Strength Index), CCI (Commodity Channel Index).



Slika 2. Grafikon EURUSD I RSI indikator

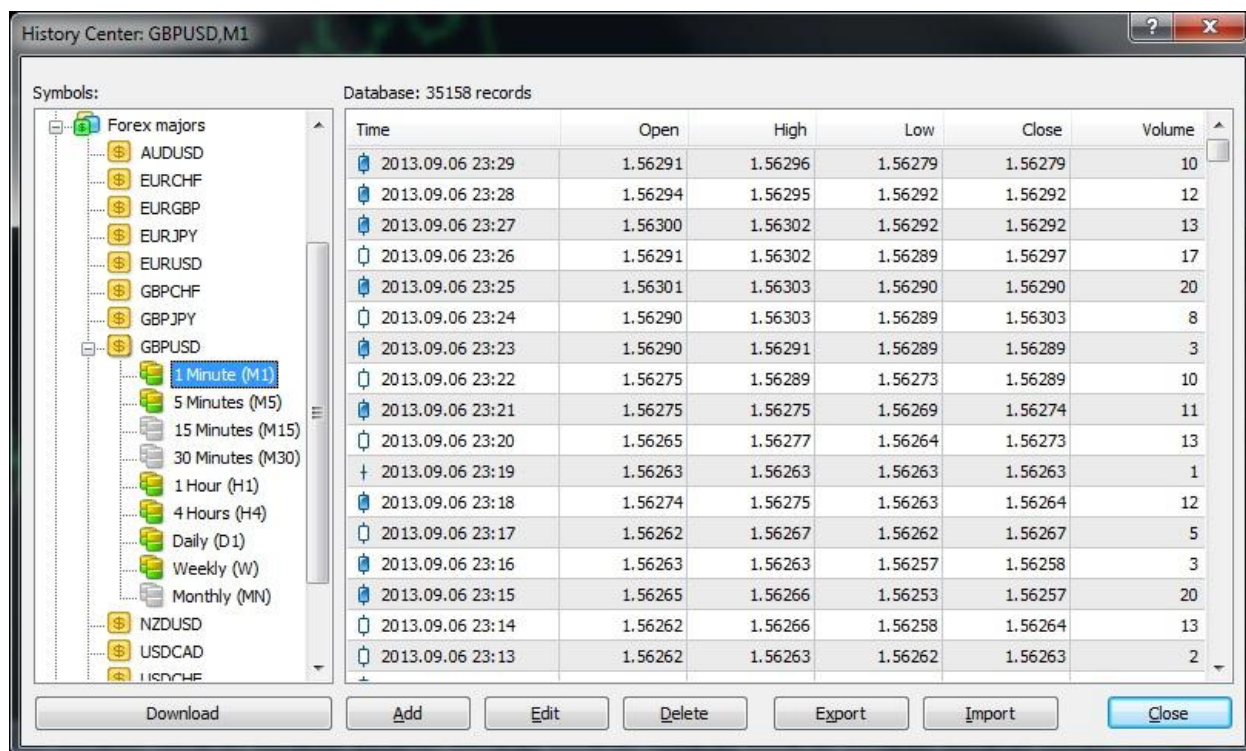
Baza podataka MetaTrader4 aplikacije je jednostavan CSV fajl u kome se cuvaju datum, vreme, cena otvaranja (OPEN), maksimalna postignuta cena (HIGH), minimalna postignuta cena (LOW), cena zatvaranja (CLOSE) i volumen.

Taj fajl se generiše kada se unutar aplikacije izabere Tools > History Center



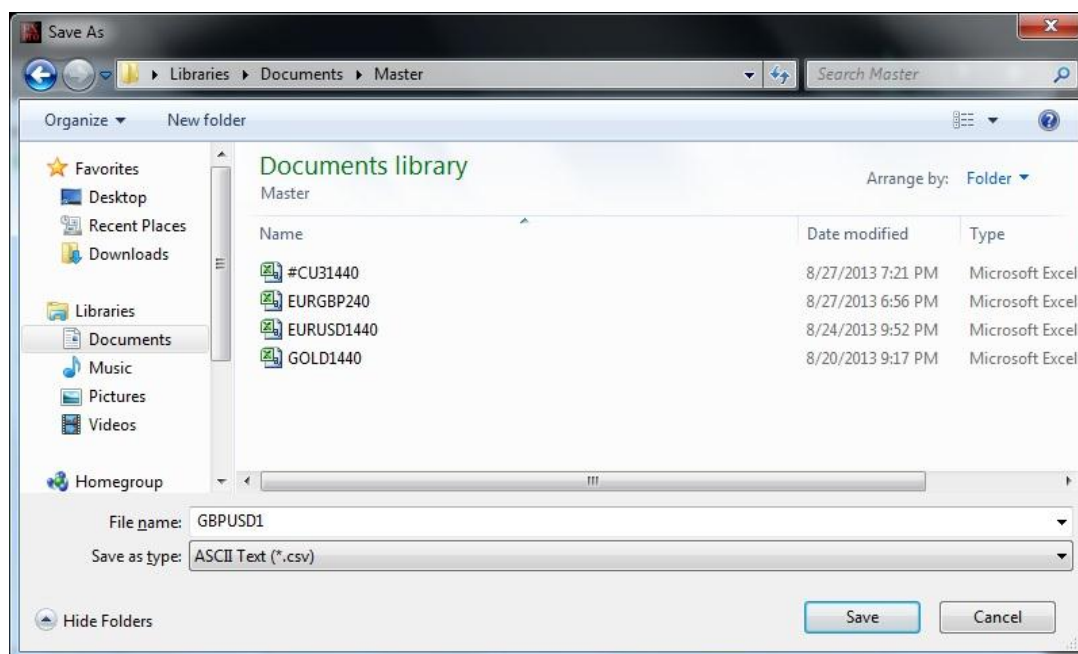
Slika 3. Aktiviranje opcije za generisanje CSV fajla

Tada se pojavljuje prozor gde korisnik bira podatke i vremenski interval za tržište koje želi da analizira.



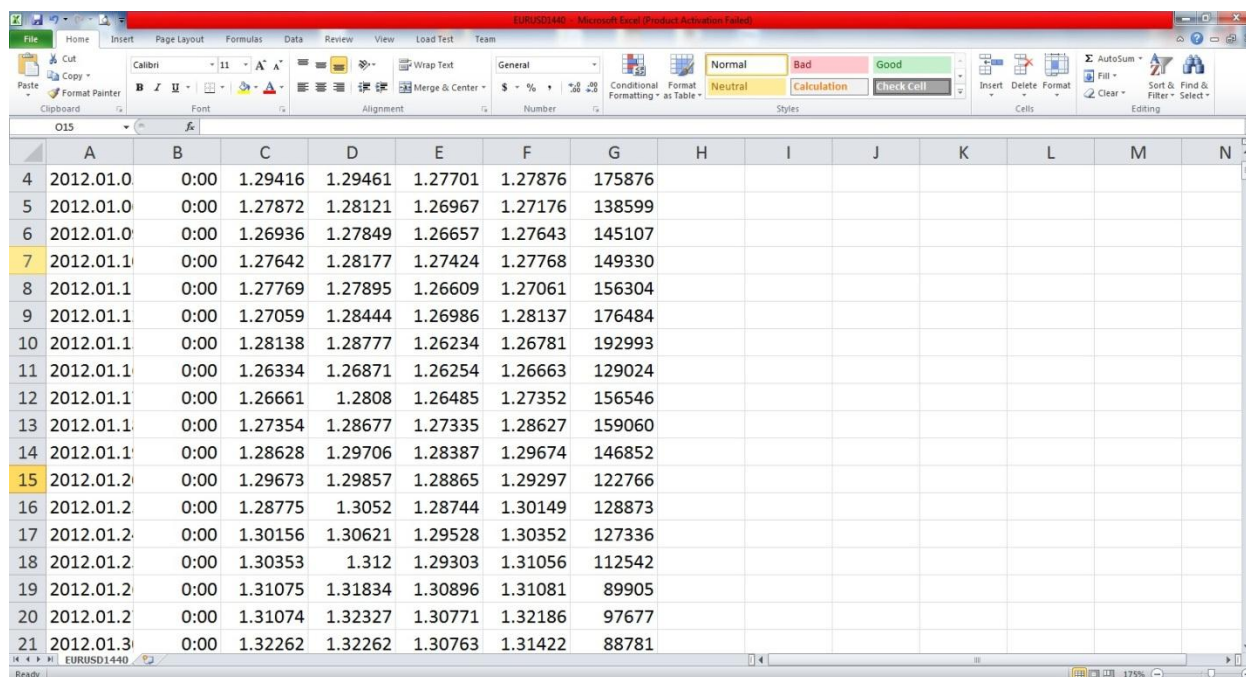
Slika 4. Izbor podataka i vremenskog intervala

Klikom na dugme Export korisnik bira gde želi da sačuva izabrane podatke.



Slika 5. Dijalog za čuvanje CSV datoteke

Sama CSV datoteka ima sledeću strukturu:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
4	2012.01.0	0:00	1.29416	1.29461	1.27701	1.27876	175876							
5	2012.01.0	0:00	1.27872	1.28121	1.26967	1.27176	138599							
6	2012.01.0	0:00	1.26936	1.27849	1.26657	1.27643	145107							
7	2012.01.1	0:00	1.27642	1.28177	1.27424	1.27768	149330							
8	2012.01.1	0:00	1.27769	1.27895	1.26609	1.27061	156304							
9	2012.01.1	0:00	1.27059	1.28444	1.26986	1.28137	176484							
10	2012.01.1	0:00	1.28138	1.28777	1.26234	1.26781	192993							
11	2012.01.1	0:00	1.26334	1.26871	1.26254	1.26663	129024							
12	2012.01.1	0:00	1.26661	1.2808	1.26485	1.27352	156546							
13	2012.01.1	0:00	1.27354	1.28677	1.27335	1.28627	159060							
14	2012.01.1	0:00	1.28628	1.29706	1.28387	1.29674	146852							
15	2012.01.2	0:00	1.29673	1.29857	1.28865	1.29297	122766							
16	2012.01.2	0:00	1.28775	1.3052	1.28744	1.30149	128873							
17	2012.01.2	0:00	1.30156	1.30621	1.29528	1.30352	127336							
18	2012.01.2	0:00	1.30353	1.312	1.29303	1.31056	112542							
19	2012.01.2	0:00	1.31075	1.31834	1.30896	1.31081	89905							
20	2012.01.2	0:00	1.31074	1.32327	1.30771	1.32186	97677							
21	2012.01.3	0:00	1.32262	1.32262	1.30763	1.31422	88781							

Slika 6. Izgled CSV datoteke otvorene pomoću Microsoft Excel-a

3. Formulacija problema

Najveći problem svakog učesnika na finansijskom tržištu jeste prognoza šta će se desiti u sledećem trenutku. Pored indikatora koji dolaze uz platformu MetaTrader4 koji olakšavaju posao tehničke analize i ekonomskih vesti koje olakšavaju fundamentalnu analizu, svaki alat koji omogućava prognoziranje je dobrodošao.

Jedan od nedostaka platforme MetaTrader4 je taj što nije omogućena primena nekog matematičkog modela na grafikon podataka u cilju prognoze budućih vrednosti.

Matematički modeli koji mogu biti interesantni za analizu tržišta, koja se svede na analizu vremenske serije, su AR(p) (eng. Autoregressive), MA(q) (eng. Moving Average), ARMA(p,q) (eng. AutoRegressive Moving Average) i ARIMA(p,d,q) (eng. AutoRegressive Integrated Moving Average) modeli gde p predstavlja red autoregresione komponente, d nivo integrisanosti vremenske serije, a q red komponente pokretnih proseka, za koje važi $p + q \leq 4$ i $d \leq 2$.

Prethodni nedostatak se može prevazići razvojem novog softverskog proizvoda ImFiStBer.

3.1. Funkcionalni zahtevi aplikacije ImFiStBer

Softver kojim se realizuje implementacija finansijskih strategija za berzanska poslovanja (tj. softver ImFiStBer) mora da obezbedi:

1. Učitavanje podataka dobijenih od strane MetaTrader4 aplikacije (učitavanje CSV fajla).
2. Implementaciju matematičkih modela AR(1), AR(2), MA(1), ARMA(1,1), ARIMA(1,1,1), ARIMA(1,2,1), ARIMA(1,1,0), ARIMA(1,2,0), ARIMA(2,1,0), ARIMA(2,2,0), ARIMA(0,1,1) i ARIMA(0,2,1) sa ciljem dobijanja poruke o formuli koja se dobija kao rezultat primene modela na učitane podatke i prognozom za sledeću vrednost u vremenskoj seriji.
3. Dicky-Foolerov test za ispitivanje stacionarnosti AR(1) vremenske serije.
4. Analiza četiri vremenske serije (High, Low, Open, Close).
5. Biranje dužine vremenske serije za analizu (podrazumevana minimalna vrednost 50).
6. Iscrtavanje grafikona podataka.
7. Iscrtavanje koeficijenata autokorelacije i parcijalne autokorelacije za učitane podatke, seriju prvih razlika i seriju drugih razlika učitanih podataka i 95% intervala poverenja.
8. GUI lak za upotrebu.

Za razvoj aplikacije izabran je programski jezik C# i razvojno okruženje Microsoft Visual Studio 2010.

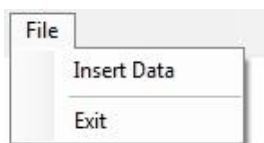
4. Grafički interfejs aplikacije ImFiStBer

Aplikacija ImFiStBer razvijena je kao GUI aplikacija u razvojnom okruženju Microsoft Visual Studio 2010 korišćenjem programskog jezika C# uz korišćenje okvira za razvoj Windows Forms.



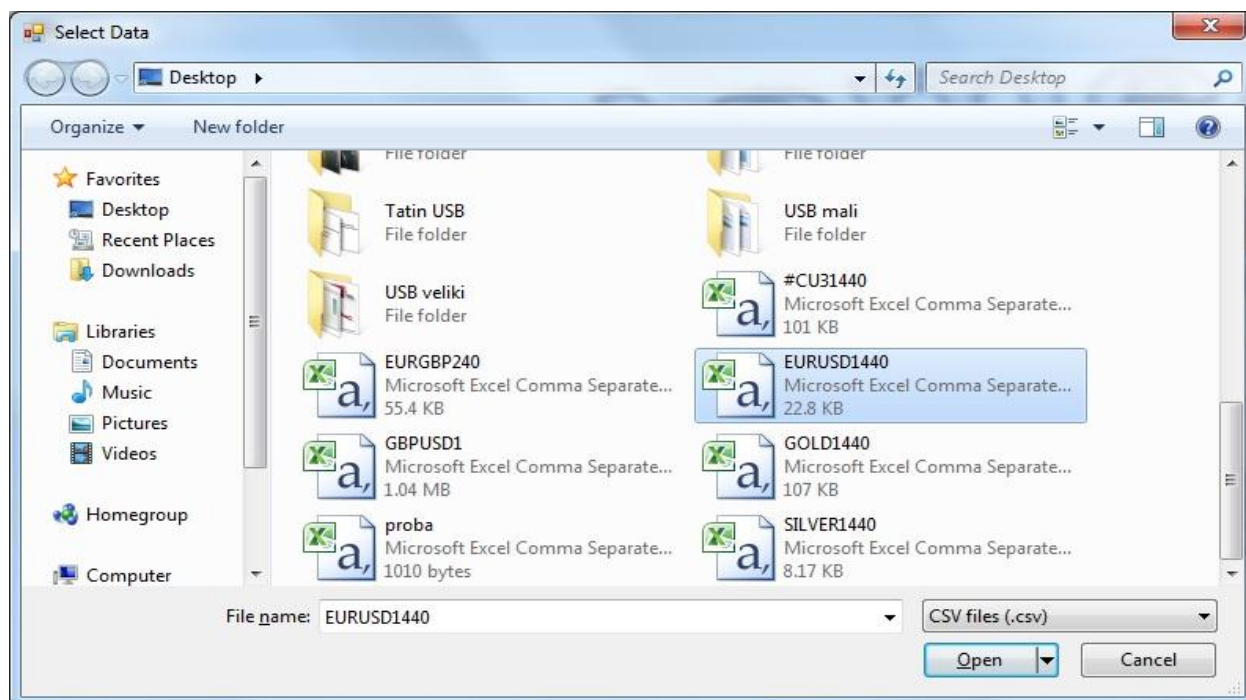
Slika 7. Izgled same aplikacije pre nego što se učitaju podaci

U File meniju korisnik ima dve mogućnosti, da učitá podatke ili da izáde iz programa.



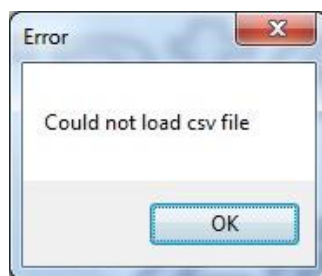
Slika 8. Izgled File menija

Ukoliko izabere da učitati podatke, pojavljuje se sledeći dijaloški prozor.



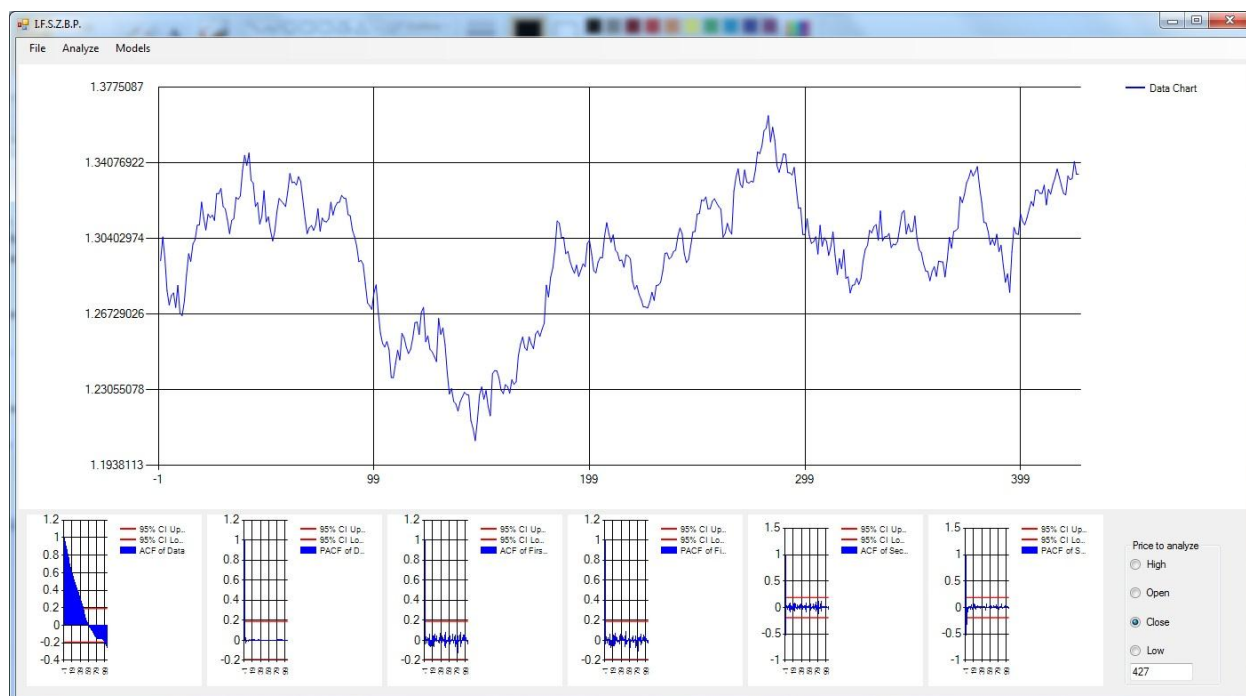
Slika9. Prozor za učitavanje podataka

Kada korisnik pokuša da učitati podatke koji nisu dobijeni od MetaTrader4 aplikacije, ili datoteku koja je dobijena od MetaTrader4 aplikacije, ali je trenutno u upotrebi od strane neke druge aplikacije (Microsoft Excel, Notepad...), prikazuje se sledeća poruka:



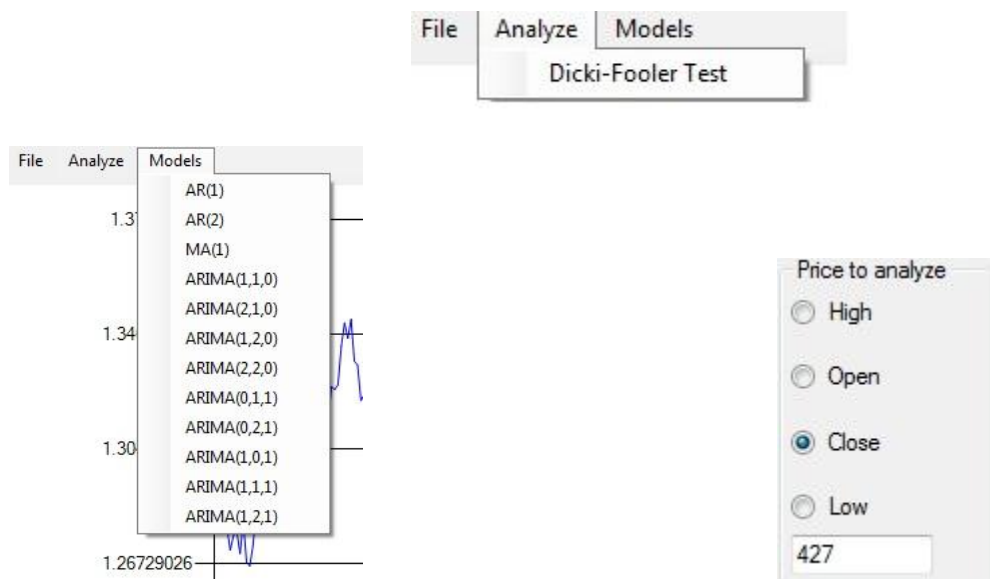
Slika 10. Poruka pri neuspešnom učitavanju

Ako nije došlo do problema prilikom učitavanja, program generiše sledeće grafikone i nove menije.



Slika 11. Program nakon učitavanja podataka (Grafikon EurUsd od 01.01.2012 do 22.08.2013)

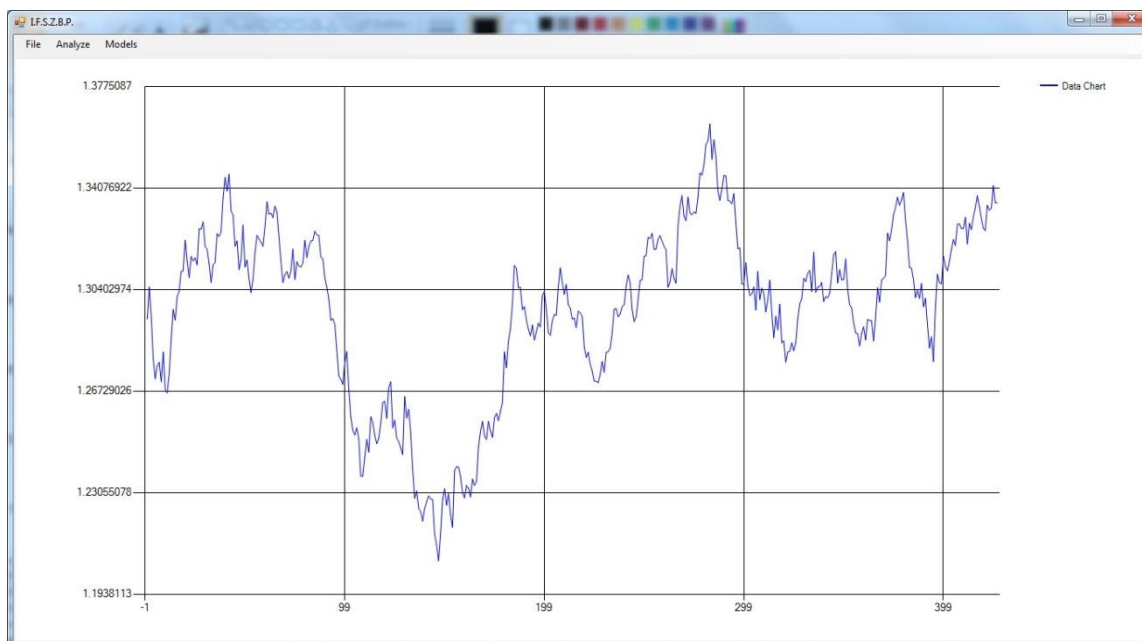
Nakon učitavanja podataka, korisniku su na raspolaganju sedam grafikona (grafikon podataka, grafikon koeficijenata autokorelacije i parcijalne autokorelacije za učitane podatke, grafikon koeficijenata autokorelacije i parcijalne autokorelacije serije prvih razlika i grafikon koeficijenata autokorelacije i parcijalne autokorelacije serije drugih razlika), dva nova padajuća menija (Analyze i Models) i Price to analyze meni.



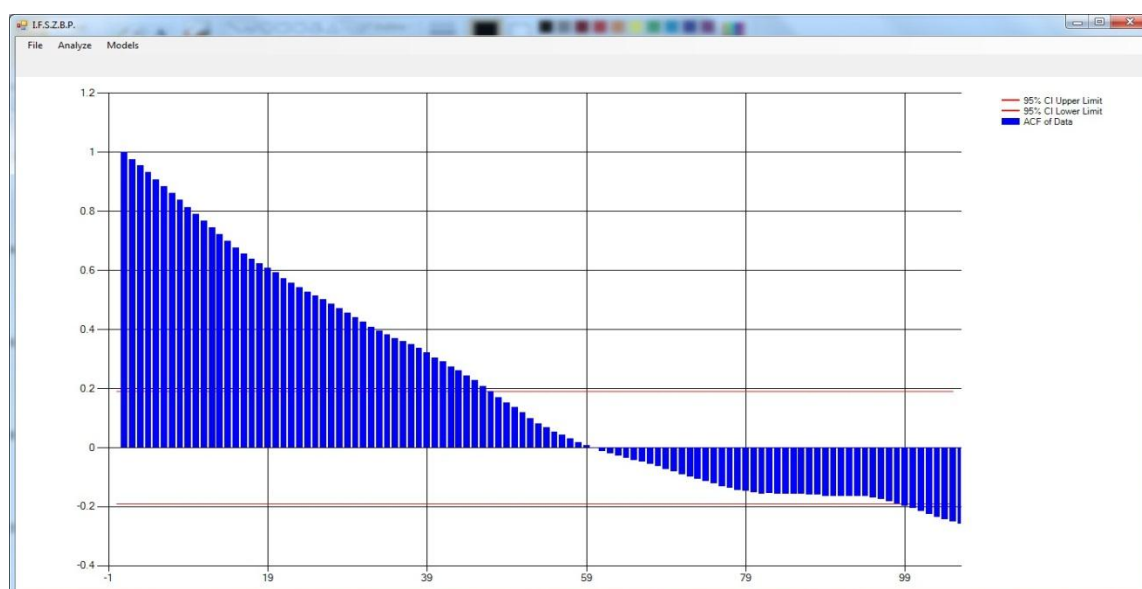
Slika 12. Opcije koje su na raspolaganju korisniku posle učitavanja CSV datoteke

Glavni deo prozora zauzima sam grafikon podataka (Data Chart). Ispod njega, redom, grafikoni koeficijenata: autokorelacije za učitane podatke (ACF of Data), parcijalne autokorelacije za učitane podatke (PACF of Data), autokorelacije prvih razlika (ACF of First Differences), parcijalne autokorelacije prvih razlika (PACF of First Differences), autokorelacije drugih razlika (ACF of Second Differences), parcijalne autokorelacije drugih razlika (PACF of Second Differences).

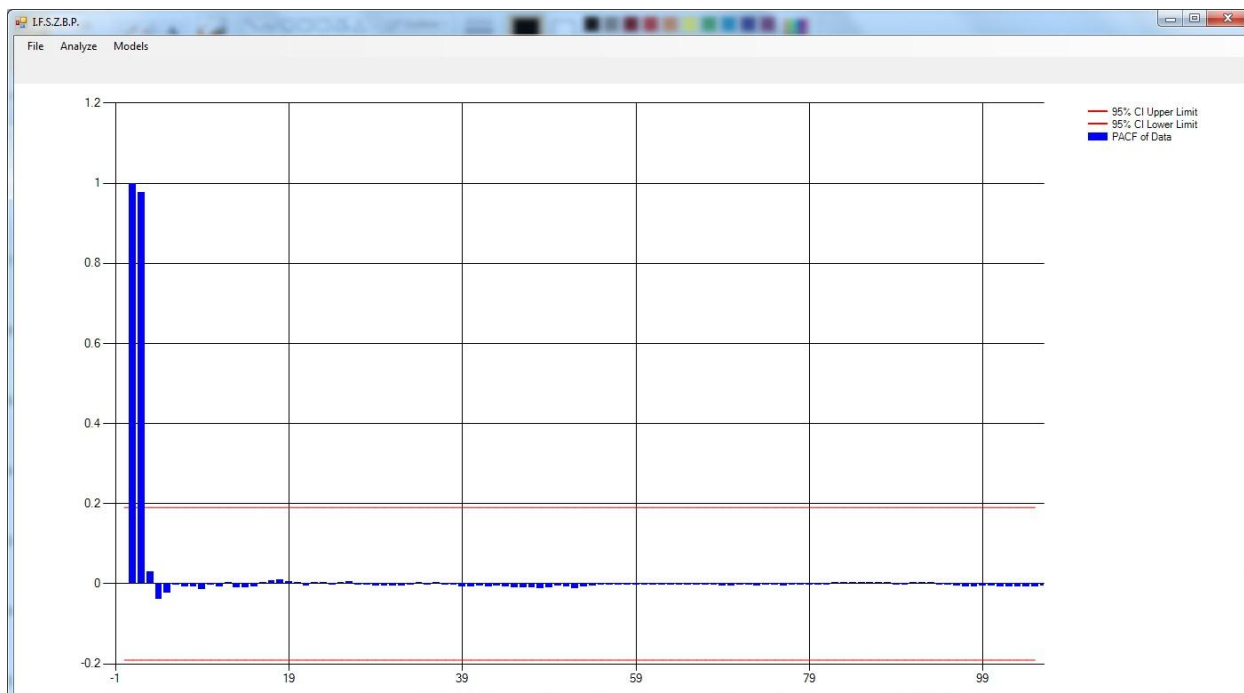
Klikom na određeni grafikon postiže se da se grafikon iscrta preko celog prozora.



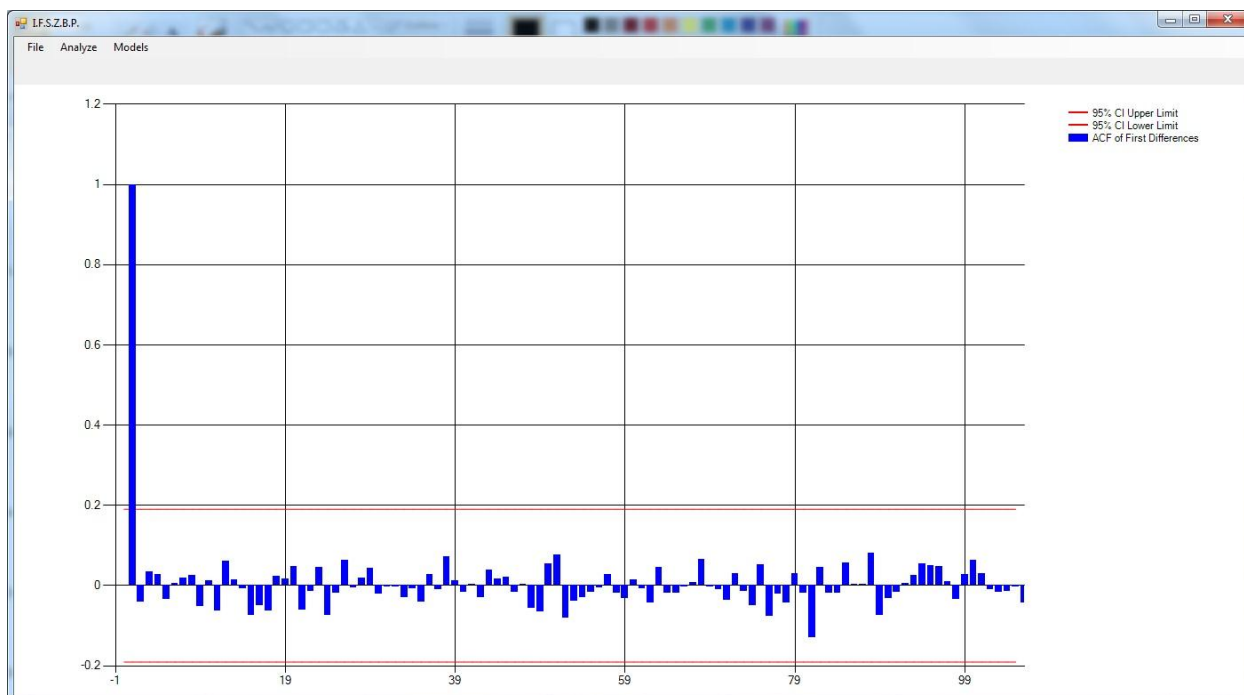
Slika 13. Grafikon podataka preko celog prozora



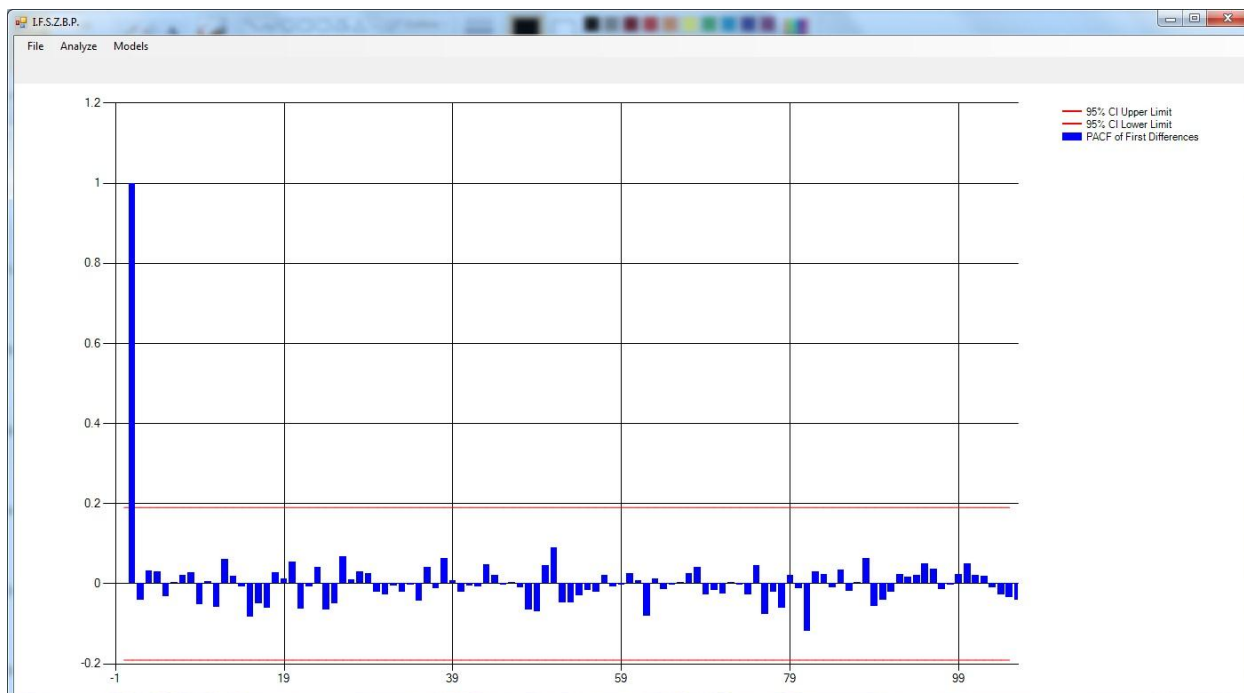
Slika 14. ACF podataka



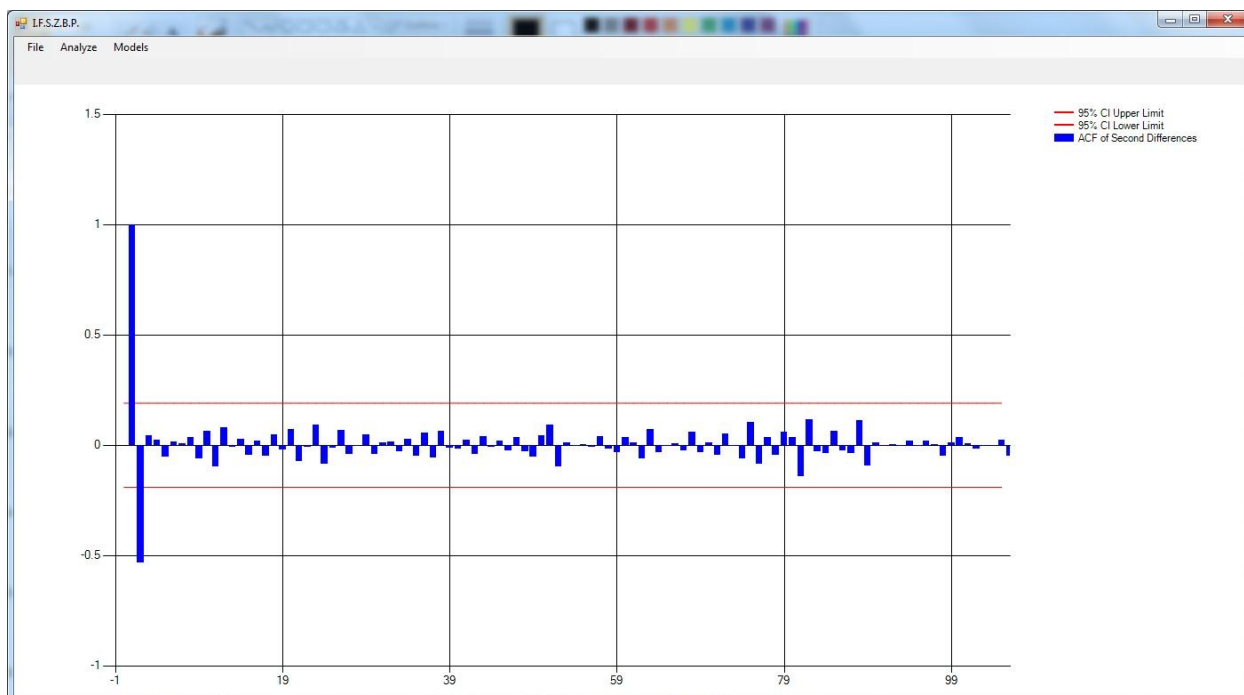
Slika 15. PACF podataka



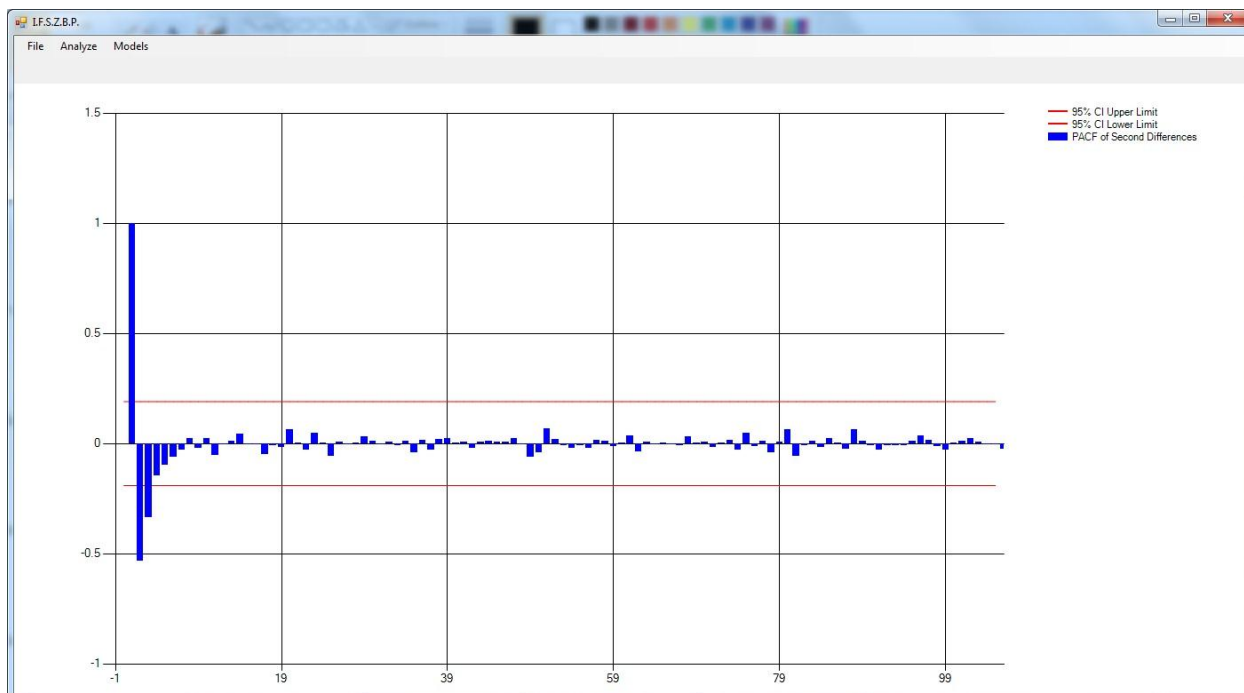
Slika 16. ACF prvih razlika



Slika 17. PACF prvih razlika



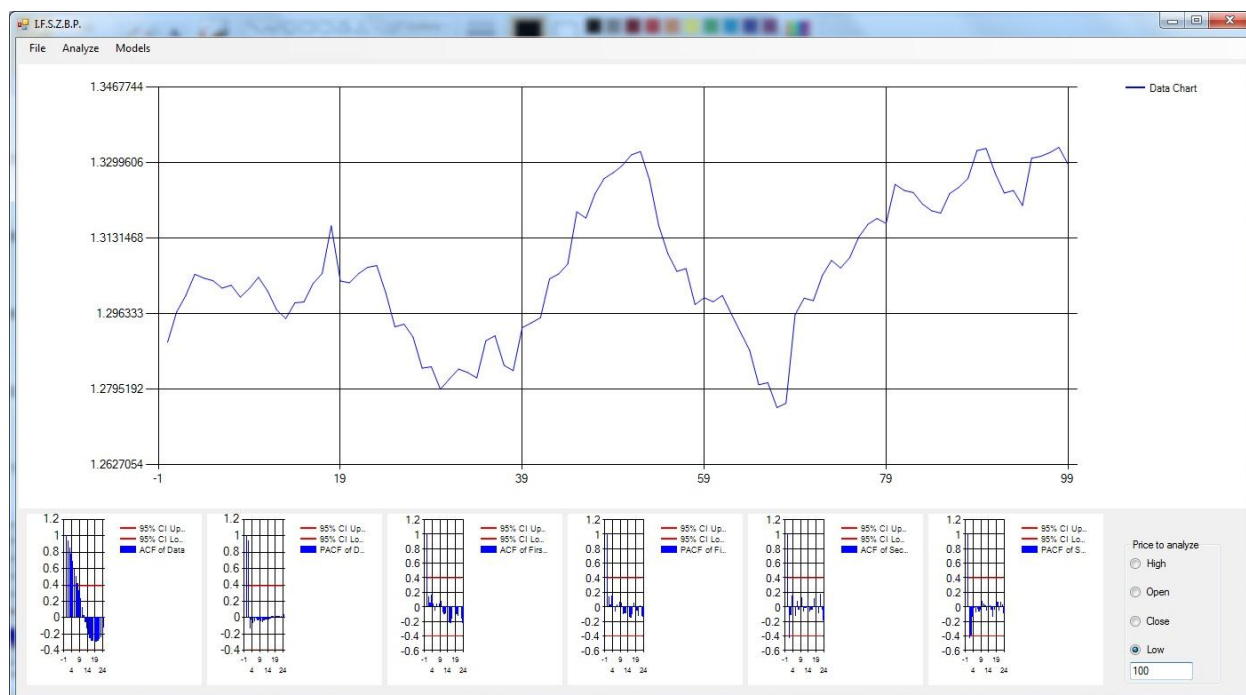
Slika 18. ACF drugih razlika



Slika 19. PACF drugih razlika

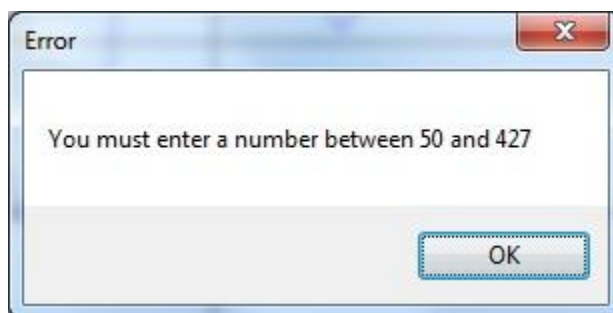
Opcija iscertavanja preko celog prozora naročito je korisna za analizu grafikona koeficijenata autokorelacije i parcijalne autokorelacije. Analizom gore navedenih grafikona, korisnik donosi odluku koji od ponuđenih modela potencijalno najbolje odgovara učitanim podacima. Ponovnim klikom na uvećani grafikon dolazi do njegovog smanjivanja i vraćanja u prvobitni položaj.

Kako učitani podaci sadrže četiri cene koje mogu da se analiziraju (Open, Low, High, Close), korisnik bira šta želi da analizira klikom na određeno radio dugme „Price to analyze“ iz menija nakon čega se automatski osvežavaju svih sedam grafikona. Podrazumevana vrednost, prilikom učitavanja samih podataka, je Close. Price to analyze meni sadrži jedno tekst polje čija je početna vrednost veličina učitanih podataka. Korisnik može tu vrednost da promeni ukoliko su učitani podaci preobimni. Najmanja vrednost koju može da unese je 50 jer ispod toga nema smisla analizirati tržište. Nakon promene automatski se osvežavaju svih sedam grafikona.



Slika 20. Isti podaci samo što se ovde analizira poslednjih 100 umesto 427.

Ukoliko korisnik pokuša da unese broj manji od 50 ili veći od veličine učitanih podataka, dolazi do prikaza sledeće poruke:



Slika 21. Poruka pri neuspešnom unosu veličine skupa podataka

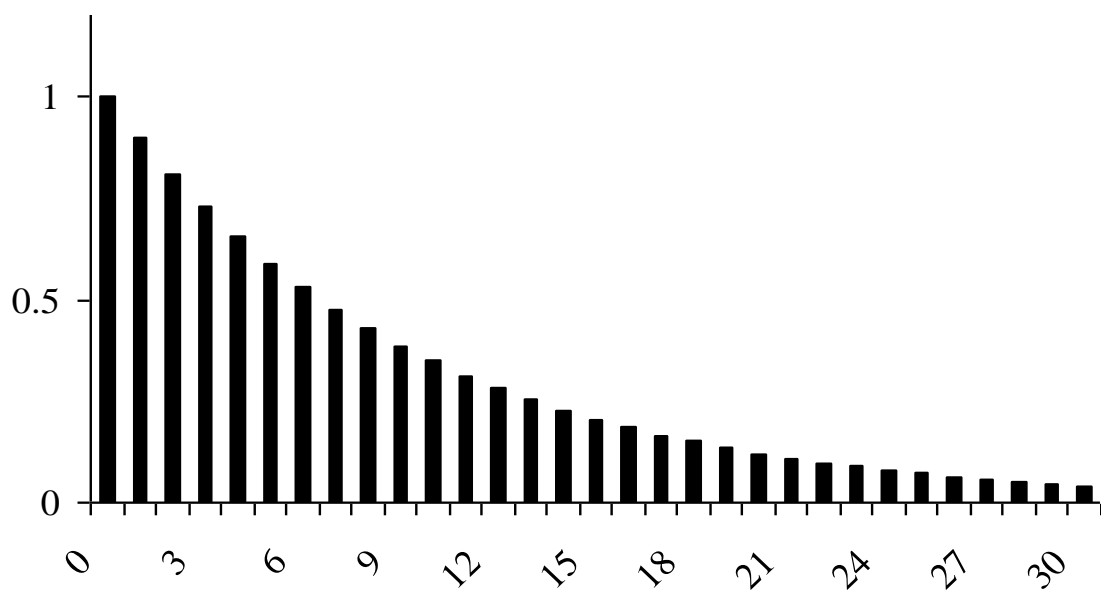
5. Implementacija strategija

U ovom poglavlju prvo se opisuje analiza grafikona koeficijenata autokorelacije i parcijalne autokorelacije koja dalje dovodi do lakšeg izbora p , d i q vrednosti odgovarajućeg $ARIMA(p,d,q)$ modela. Zatim se analizira stacionarnost $AR(1)$ vremenske serije (test jediničnog korena (Dickey-Fuller test)). Potom se opisuju četiri osnovna modela ($AR(1)$, $AR(2)$, $MA(1)$, $ARIMA(1,0,1)$) koji se koriste za izračunavanje ostalih modela ($ARIMA(1,1,1)$, $ARIMA(1,2,1)$, $ARIMA(1,1,0)$, $ARIMA(1,2,0)$, $ARIMA(0,1,1)$, $ARIMA(0,2,1)$, $ARIMA(2,1,0)$, $ARIMA(2,2,0)$).

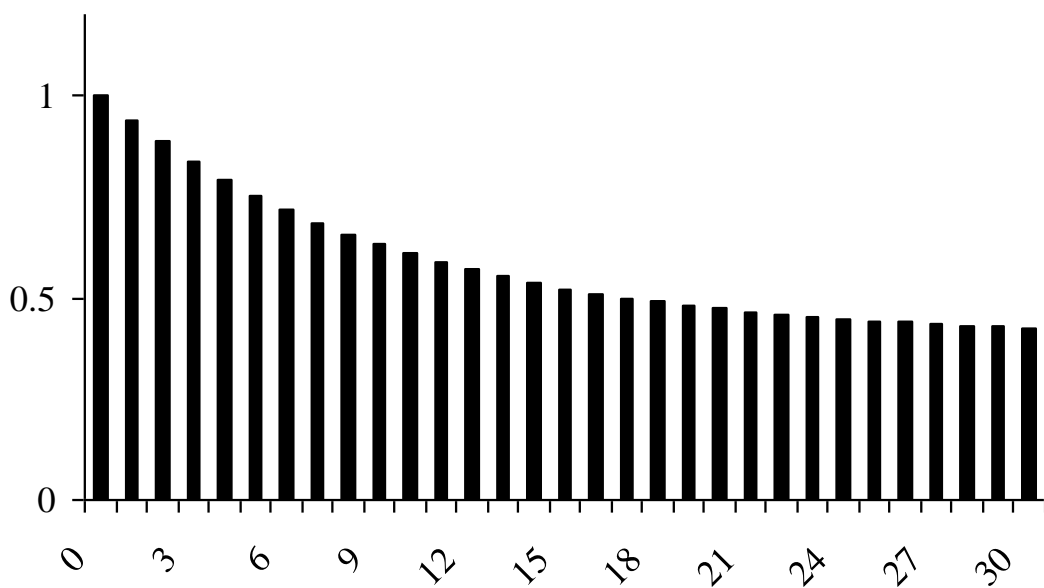
5.1. Analiza grafikona koeficijenata autokorelacije i parcijalne autokorelacije. Izbor p,d,q vrednosti.

Osobine ACF i PACF funkcija za MA, AR i ARMA serije			
Proces	MA(q)	AR(p)	ARMA(p,q)
ACF	Vrednosti opadaju tokom vremena i jednake su nuli za pozicije veće od reda procesa q .	Vrednosti opadaju tokom vremena po eksponencijalnoj ili sinusoidnoj putanji	Vrednosti opadaju tokom vremena. Prvih p koeficijenata je određeno parametrima AR i MA komponente, dok za pozicije veće od p koeficijenti zavise samo od MA komponente
PACF	Vrednosti opadaju tokom vremena po eksponencijalnoj ili sinusoidnoj putanji	Vrednosti su različite od nule samo za pozicije koje su manje ili jednake redu procesa p	Vrednosti opadaju tokom vremena. Prvih q koeficijenata je određeno parametrima AR i MA komponente, dok za pozicije veće od q koeficijenti zavise samo od AR komponente

Tabela 1. Osobine ACF i PACF funkcija [1][2][4][5]



Slika 22. Primer ACF koeficijenata stacionarne serije



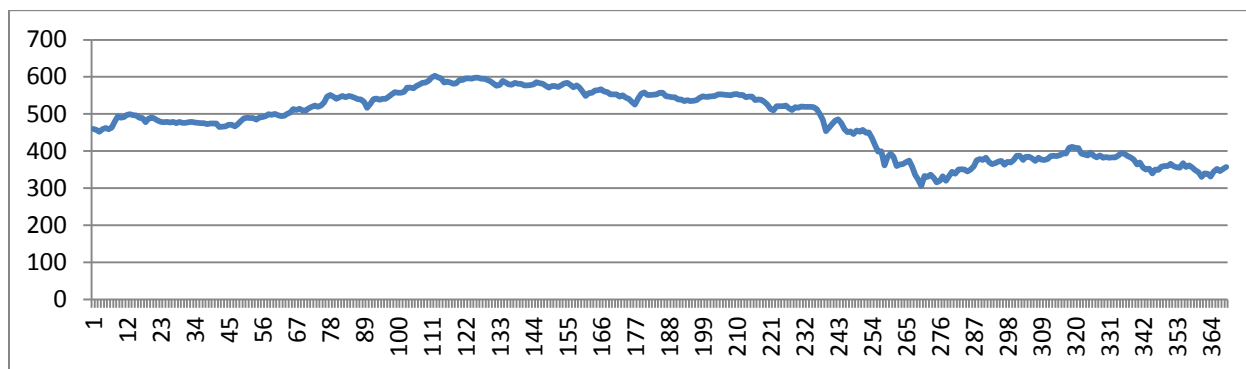
Slika 23. Primer ACF koeficijenata nestacionarne serije

Parametar d se bira kada se analizom grafikona koeficijenata autokorelacije utvrdi da je posmatrana serija stacionarna. Tek kada se odredi parametar d prelazi se na određivanje parametara p i q [2][6][7].

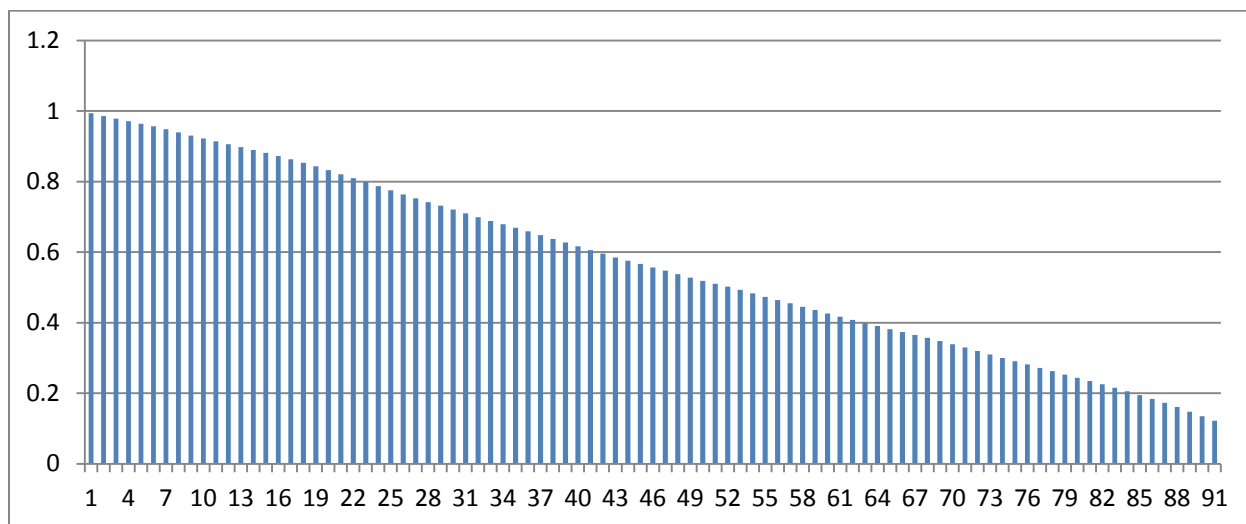
Vrednost p se bira iz gore navedene tabele posmatranjem koeficijenta parcijalne autokorelacije stacionarne serije.

Vrednost q se bira iz gore navedene tabele posmatranjem koeficijenta autokorelacije stacionarne serije.

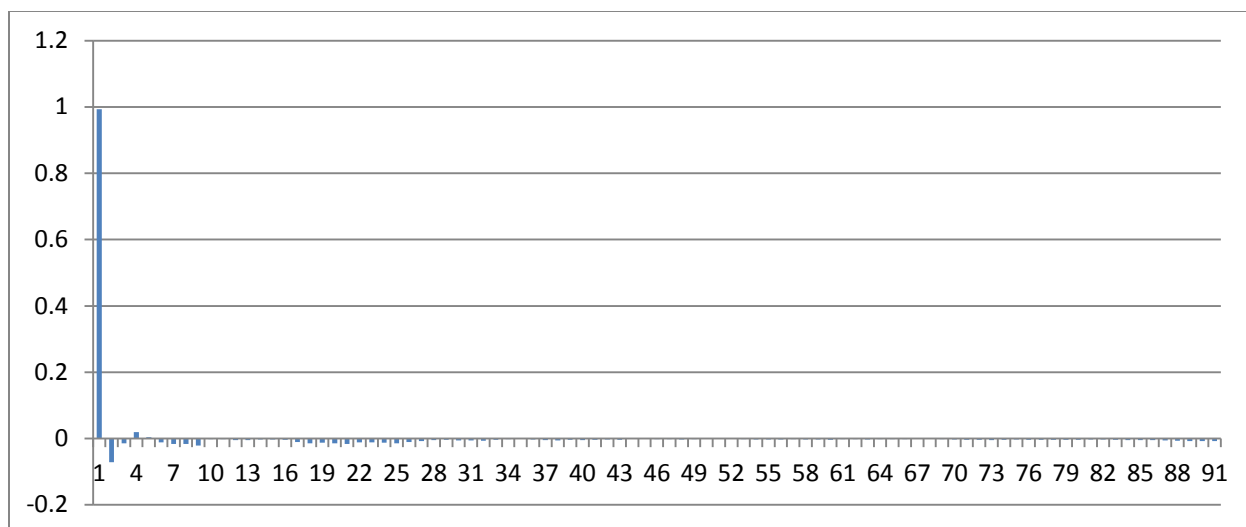
Primer:



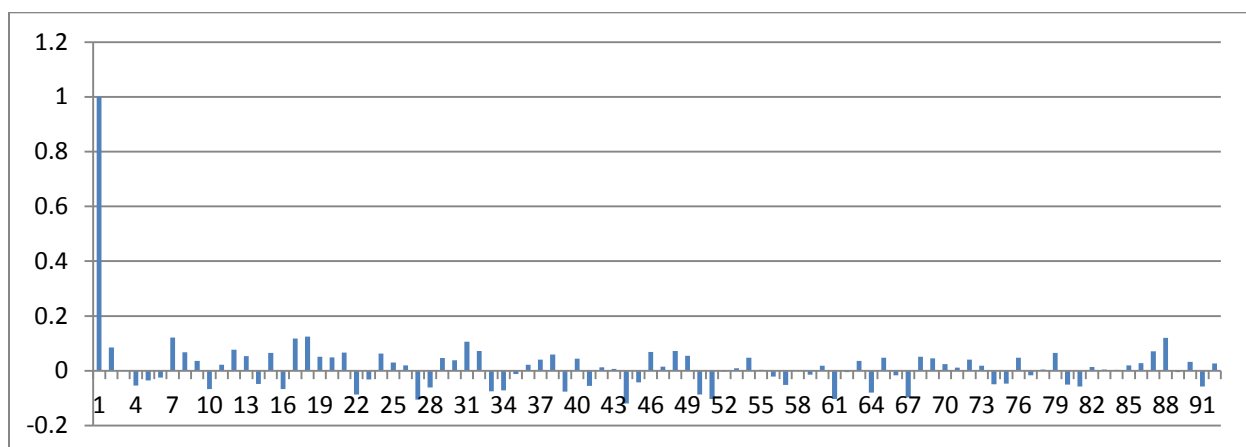
Slika 24. Vrednost akcija kompanije IBM od 17.05.1961 do 02.11.1962



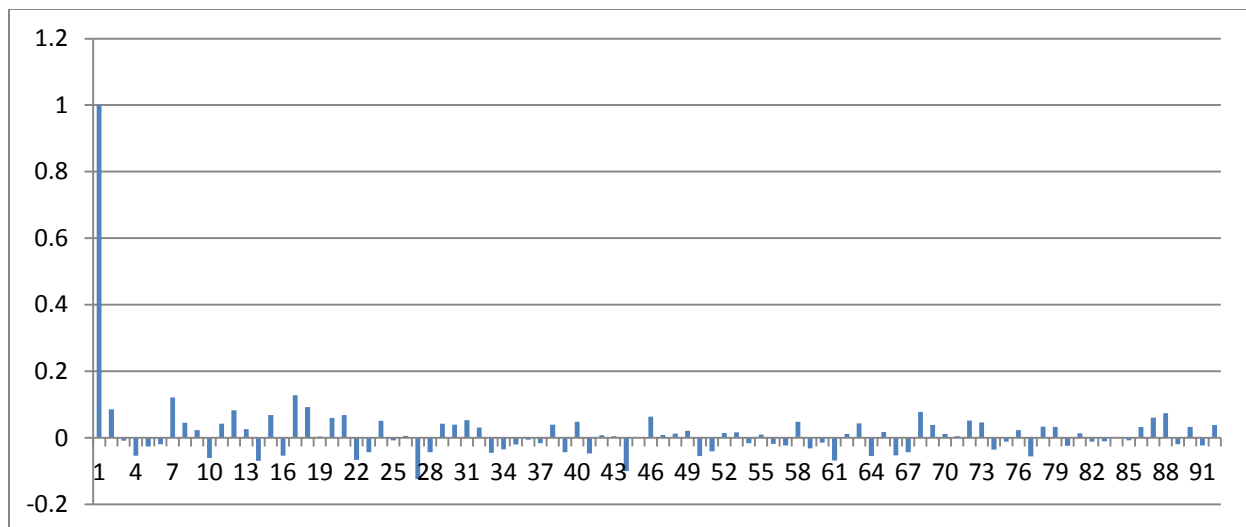
Slika 25. Grafikon koeficijenta autokorelacije



Slika 26. Grafikon koeficijenata parcijalne autokorelacije



Slika 27. Grafikon koeficijenata autokorelacije prvih razlika



Slika 28. Grafikon koeficijenta parcijalne autokorelacije prvih razlika

Analizom grafikona se zaključuje da je $d=1$.

Da bismo odredili parametre p i q , potrebno je izračunati 95% interval poverenja. On se računa po formuli:

$$\pm \frac{2}{\sqrt{T}}$$

gde T predstavlja dužinu analizirane serije [3].

U ovom slučaju $T=92$ pa se zaključuje da je 95% interval poverenja 0,208, iz čega se dalje zaključuje da grafikon koeficijenta autokorelacije i parcijalne autokorelacije predstavljaju beli šum, tj. na njih se gleda kao da su jednaki nuli. Dakle, vrednosti parametara za IBM podatke su:

$$p=0 \quad d=1 \quad q=0$$

5.2. Dicky-Fooler Test. Ispitivanje stacionarnosti AR(1) vremenske serije.

Dicky-Foolerovim testom (ili test jediničnog korena) ispituje se da li je vremenska serija stacionarna ili poseduje jedinične korene (Jedinični koren je karakteristika vremenske serije koja može da dovede do problema u statističkom zaključivanju koji matematički model je prikladan za tu vremensku seriju) [1].

Pretpostavlja se da je vremenska serija X_t generisana AR(1) modelom:

$$X_t = \beta_0 + \varphi * X_{t-1} + e_t$$

$$\beta_0 = \mu * (1 - \varphi)$$

Priroda kretanja X_t zavisi od parametra φ : za $\varphi < 1$, X_t sledi stacionarnu putanju, dok za $\varphi = 1$, X_t poseduje jedinični koren. Ovo tvrđenje se može iskoristiti za posatvljanje sledeće hipoteze:

H_0 : $\varphi = 1$ (X_t poseduje jedinicni koren)

H_1 : $\varphi < 1$ (X_t je stacionarna vremenska serija oko nenulte srednje vrednosti)

Vrednosti nulte hipoteze se ispituje na osnovu primene odgovarajuće test statistike koja se računa prema podacima iz uzorka obima T : X_1, X_2, \dots, X_t .

Prvo se ocenjuje parameter φ primenom metode običnih najmanjih kvadrata

$$\varphi = \frac{\sum_{t=2}^T (X_t - \bar{X}) * (X_{t-1} - \bar{X})}{\sum_{t=2}^T (X_{t-1} - \bar{X})^2}$$

Potom se računa standardna greška ocene $s(\varphi)$ na sledeći način:

$$s(\varphi) = \sqrt{\frac{s^2}{\sum_{t=2}^T (X_{t-1} - \bar{X})^2}} \quad S^2 = \frac{\sum_{t=2}^T (X_t - \bar{X})^2 - \varphi * \sum_{t=2}^T (X_t - \bar{X}) * (X_{t-1} - \bar{X})}{(T-1)-1}$$

Imajući u vidu teoriju statističkog zaključivanja, može se obrazovati test statistika:

$$\frac{\varphi}{s(\varphi)}$$

koja bi imala t-raspodelu. Ukoliko bi bila tačna nulta hipoteza, tj. ocena da vremenska serija poseduje jedinični koren, tada ocena φ nema normalnu raspodelu, što znači da količnik $\frac{\varphi-1}{s(\varphi)}$ nema t-raspodelu. Dicky i Fuller su prvi pokazali da je raspodela datog količnika nestandardna jer ne pripada klasi poznatih raspodela. Ova raspodela se označava τ -raspodela ili DF test jediničnog korena. Možemo pisati:

$$\tau_\mu = \frac{\varphi}{s(\varphi)}$$

Kritična vrednost τ_μ^t se može odrediti za svaki obim uzorka upotrebom određene formule, koja za nivo značajnosti 5% glasi:

$$\tau_\mu^t = -2,8621 - \frac{2,738}{T} - \frac{8,36}{T^2}$$

Funkcija koja računa φ :

```
public double fi(double[] x)
{
    int n = x.Length;
    double x_mean = x.Average();

    double sum1 = 0;
    double sum2 = 0;
```

```

    double quotient = 0;

    for (int i = 1; i < n; i++)
    {
        sum1 += (x[i] - x_mean) * (x[i - 1] - x_mean);
        sum2 += (x[i - 1] - x_mean) * (x[i - 1] - x_mean);
    }
    quotient = sum1 / sum2;

    return quotient;
}

```

Funkcija koja računa S^2 :

```

public double s_squared(double fi, double[] x)
{
    double s_squared = 0;

    double x_mean = x.Average();
    int n = x.Length;
    double sum1 = 0;
    double sum2 = 0;

    for (int i = 1; i < n; i++)
    {
        sum1 += (x[i]-x_mean) * (x[i]-x_mean);
        sum2 += (x[i]-x_mean) * (x[i - 1]-x_mean);
    }

    s_squared = (sum1 - fi * sum2) / ((n - 1) - 1);

    return s_squared;
}

```

Funkcija koja računa $s(\varphi)$:

```

public double estimate_s(double sna2, double[] x)
{
    double estimate = 0;

    int n = x.Length;
    double x_mean = x.Average();
    double sum1 = 0;
    double quotient;

    for (int i = 1; i < n; i++)
    {
        sum1 += (x[i - 1] - x_mean) * (x[i - 1] - x_mean);
    }

    quotient = sna2 / sum1;

    estimate = Math.Sqrt(quotient);
    return estimate;
}

```

```
}
```

Funkcija koja računa test statistiku τ_{μ} :

```
public double test_statistics(double fi, double estimate)
{
    double test = 0;

    test = fi / estimate;

    return test;
}
```

Funkcija koja računa kritičnu vrednost τ_{μ}^t :

```
public double critical(int T)
{
    double crit = 0;

    crit = -2.8621 - (2.738 / T) - (8.36 / (T * T));
    return crit;
}
```

Funkcija koja generiše string sa rezultatima Dicky-Foolerovog testa:

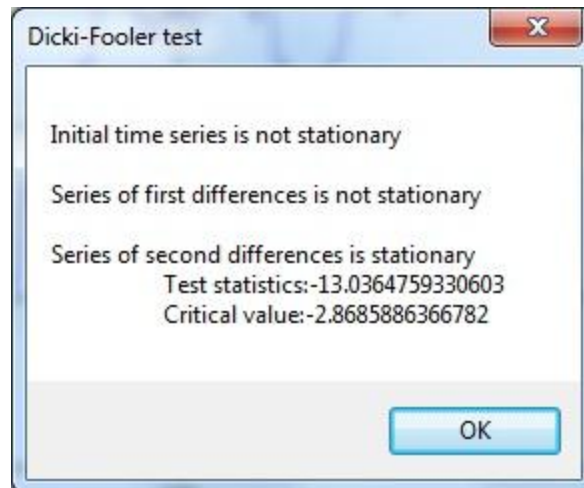
```
public string DickyFooler(double[] x, int k, string s)
{
    string msg = string.Empty;

    double[] val = TrimArray(x, k);

    double Fi = fi(val);
    double s2 = s_squared(Fi, val);
    double ocenaS = estimate_s(s2, val);
    double test = test_statistics(Fi, ocenaS);
    double crit = critical(val.Length);

    if (test < crit)
        msg = string.Format("{0} is stationary \n\tTest statistics:{1}\n\tCritical value:{2}", s, test.ToString(), crit.ToString());
    else
        msg = string.Format("{0} is not stationary ", s);

    return msg;
}
```



Slika 29. Poruka sa rezultatima Dicki-Fooler testa

5.3. Modeli

Glavni modeli potrebni za izvođenje svih ostalih su: AR(1), AR(2), MA(1), ARIMA(1,0,1). Model AR(1) se koristi za dobijanje modela ARIMA(1,1,0) i ARIMA(1,2,0), AR(2) za dobijanje modela ARIMA(2,1,0) i ARIMA(2,2,0), MA(1) za dobijanje modela ARIMA(0,1,1) i ARIMA(0,2,1), dok se ARIMA(1,0,1) model koristi za dobijanje ARIMA(1,1,1) i ARIMA(1,2,1) modela. Glavna razlika između osnovnih i izvedenih modela je u vrednostima koje se prosleđuju, a koje se koriste za računanje prognozirane vrednosti.

5.3.1. AR(1)

Def. Neka je $\{\varepsilon_t\}$ niz nekorelisanih $(0; \sigma^2)$ slučajnih promenljivih i β realan parameter. Za seriju $\{X_t\}$ definisanu na sledeći način:

$$X_t = \beta X_{t-1} + \xi_t, \quad t \in (0, \pm 1, \pm 2, \pm 3, \dots).$$

kaže se da je autoregresijska serija prvog reda (AR(1) serija) [2].

Prognoza za AR(1) seriju:

$$X_{n+1} = E(X_{n+1} | X_n, X_{n-1}, X_{n-2}, X_{n-3}, \dots) = \beta X_n + E \xi_{n+1} = \beta X_n$$

Ocena parametra β :

$$\beta = \rho_1$$

```

public string AR1(double[] Values, double[] Coefs, int k)
{
    string msg = string.Empty;

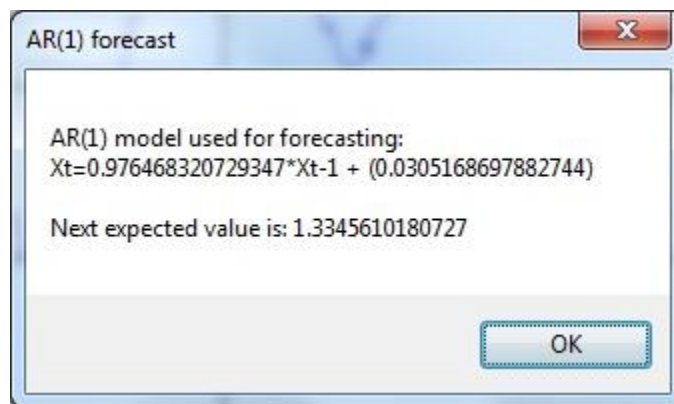
    double[] val = TrimArray(Values, k);

    double x_mean = val.Average();
    int n = val.Length;
    double delta = x_mean * (1 - Coefs[1]);

    string model = string.Format("Xt={0}*Xt-1 + ({1})", Coefs[1], delta);
    double forecast = Coefs[1] * val[n - 1] + delta;
    msg = string.Format("AR(1) model used for forecasting:\n" + model + "\n\nNext  
expected value is: "+forecast.ToString());

    return msg;
}

```



Slika 30. Poruka sa rezultatima AR(1) prognoze

5.3.2. AR(2)

Def. Neka je $\{\varepsilon_t\}$ niz nekorelisanih $(0; \sigma^2)$ slučajnih promenljivih i b_1 i b_2 realni parametri. Za seriju $\{X_t\}$ definisanu na sledeći način:

$$X_t = b_1 X_{t-1} + b_2 X_{t-2} + \varepsilon_t, \quad t \in (0, \pm 1, \pm 2, \pm 3, \dots).$$

kaže se da je autoregresijska serija drugog reda (AR(2) serija) [2].

Prognoza za AR(2) seriju:

$$X_{n+1} = E(X_{n+1} / X_n, X_{n-1}, X_{n-2}, X_{n-3}, \dots) = b_1 X_n + b_2 X_{n-1}$$

Ocena parametara b_1 i b_2 :

b_1 i b_2 dobijamo kao rešenja Džul-Vokerovih jednačina za dve nepoznate.

$$1*b_1 + \rho_1*b_2 = \rho_1$$

$$\rho_1*b_1 + 1*b_2 = \rho_2$$

```
public string AR2(double[] Values, double[] Coefs, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    double r1 = Coefs[1];
    double r2 = Coefs[2];

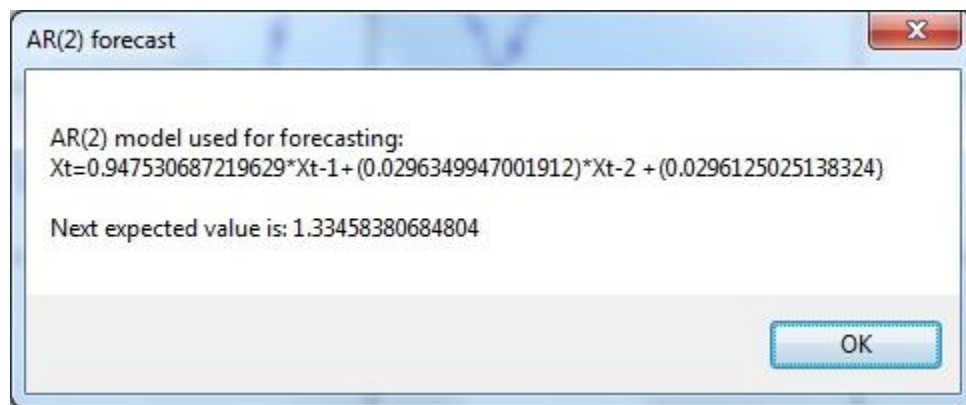
    double beta1 = (r1 - r1 * r2) / (1 - r1 * r1);
    double beta2 = (r1 - beta1) / r1;

    double x_mean = val.Average();
    int n = val.Length;
    double delta = x_mean * (1 - beta1 - beta2);

    string model = string.Format("Xt={0}*Xt-1+({1})*Xt-2 + ({2})", beta1, beta2,
                                delta);

    double forecast = beta1 * val[n - 1] + val[n - 2] * beta2 + delta;
    msg = string.Format("AR(2) model used for forecasting: \n" + model + "\n\nNext
    expected value is: " + forecast.ToString());

    return msg;
}
```



Slika 31. Poruka sa rezultatima AR(2) prognoze

5.3.3. MA(1)

Def. Neka je $\{\varepsilon_t\}$ niz nekorelisanih $(0; \sigma^2)$ slučajnih promenljivih i a realan parameter. Za seriju $\{X_t\}$ definisanu na sledeći način:

$$X_t = \xi_t + a \xi_{t-1}, \quad t \in (0, \pm 1, \pm 2, \pm 3, \dots).$$

kaže se da je serija pokretnih proseka prvog reda (MA(1) serija) [2].

Prognoza za MA(1) seriju:

$$X_{n+1} = E(X_{n+1} / \xi_t, \xi_{t-1}, \xi_{t-2}, \xi_{t-3}, \dots) = E \xi_{n+1} + a \xi_n = a \xi_n$$

Uslov koji mora biti ispunjen da bi se model MA(1) uzeo u razmatranje je:

$$-0.5 < \rho_1 < 0.5, \quad (\rho_1 \text{ prvi koeficijent autokorelacione funkcije})$$

Ocena parametra a :

$$\rho_1 = \frac{a}{1 + a^2}$$
$$a = \frac{1}{2\rho_1} \pm \sqrt{\frac{1}{4\rho_1} - 1}$$

```
public string MA1(double[] Values, double[] Coefs, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    double r1 = Coefs[1];
    if (Math.Abs(r1) > 0.5)
        msg = "MA(1) model is not appropriate for given time series";
    else
    {
        double alfa = 0;

        double alfa1 = (1 - Math.Sqrt((1 - 4 * r1 * r1))) / (2 * r1);
        double alfa2 = (1 + Math.Sqrt((1 - 4 * r1 * r1))) / (2 * r1);

        if (Math.Abs(alfa1) < 1)
            alfa = alfa1;
        else if (Math.Abs(alfa2) < 1)
            alfa = alfa2;
        else
            alfa = 0;

        if (alfa != 0)
        {
            double x_mean = val.Average();
            int n = val.Length;
```

```

double[] u = new double[n];

for (int i = 0; i < n; i++)
{
    u[i] = val[i] - x_mean;
}

//Ut-1 = Xt-1 - X_prog_t-1
string model = string.Format("Xt={0}*Ut-1 + ({1})", alfa, x_mean);

double forecast = alfa * u[n - 1] + x_mean;
msg = string.Format("MA(1) model used for forecasting:\n" + model + "\n\n"
    Next expected value is: " + forecast.ToString());
}
}

return msg;
}

```



Slika 32. Poruka sa rezultatima MA(1) prognoze

5.3.4. ARMA(1,1) = ARIMA(1,0,1)

Def. Serija $\{X_t, t \in (0, \pm 1, \pm 2, \pm 3, \dots)\}$ relanih slučajnih veličina definisana sa

$$\sum_{j=0}^p b_j X_{t-j} = \sum_{v=0}^q a_v \varepsilon_{t-v} \quad , t \in D$$

u kojoj će se smatrati da je $b_0=1$, $b_p \neq 0$, $a_0=1$, $a_q \neq 0$ i gde je $\{\varepsilon_t\}$ niz nekorelisanih $(0; \sigma^2)$ slučajnih promenljivih je jedna autoregresijska serija pokretnih proseka reda (p,q) (ARMA(p,q)) [2].

Prognoza za ARMA(1,1) seriju:

$$X_{n+1} = bX_n + a\xi_n$$

Ocene parametara a i b:

$$\rho_1 = \frac{(1+ab)(a+b)}{1+a^2+2ab}$$

$$\rho_2 = \rho_1 b$$

```

public string ARIMA101(double[] Values, double[] Coefs, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    double r1 = Coefs[1];
    double r2 = Coefs[2];

    double x_mean = val.Average();

    double beta = r2 / r1;

    double delta = x_mean * (1 - beta);

    double a, c;
    a = c = r1 - beta;
    double b = (2 * r2) - 1 - (beta * beta);

    double alfa1, alfa2, alfa;

    alfa1 = ((-1) * b + Math.Sqrt((b * b - 4 * a * c))) / (2 * a);
    alfa2 = ((-1) * b - Math.Sqrt((b * b - 4 * a * c))) / (2 * a);

    if (Math.Abs(alfa1) < 1)
        alfa = alfa1;
    else if (Math.Abs(alfa2) < 1)
        alfa = alfa2;
    else
        alfa = 0;

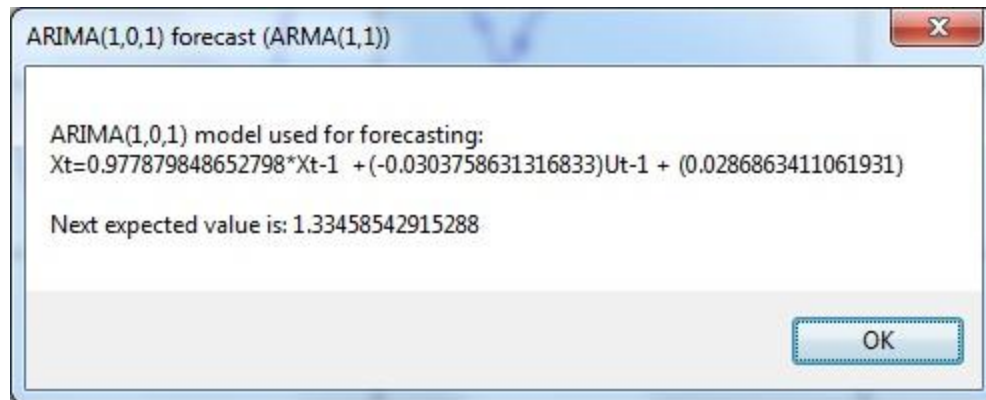
    if (alfa != 0)
    {
        int n = val.Length;
        double[] u = new double[n - 1];

        for (int i = 1; i < n; i++)
        {
            u[i - 1] = val[i] - beta * val[i - 1] - delta;
        }

        string model = string.Format("Xt={0}*Xt-1 +({1})Ut-1  
+ ({2})", beta, alfa, delta);
        double forecast = beta * val[n - 1] + alfa * u[n - 2] + delta;
        msg=string.Format("ARIMA(1,0,1) model used for forecasting:\n" + model +  
"\n\nNext expected value is: " + forecast.ToString());
    }
    else
        msg = "ARIMA(1,0,1) model is not appropriate for given time series";

    return msg;
}

```



Slika 33. Poruka sa rezultatima ARIMA(1,0,1) prognoze

5.3.5. ARIMA(1,1,1) i ARIMA(1,2,1)

ARIMA(1,1,1) se računa tako što se funkciji ARIMA(1,0,1) proslede vrednosti niza serije prvih razlika i koeficijenti autokorelacije prvih razlika umesto učitano g niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija ARIMA(1,0,1) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju prvih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(1,1,1).

```
internal string ARIMA111(double[] Values, double[] FirstDifferences, double[] Coefs_FD, int k)
```

```
{
    string msg = string.Empty;
    double[] val = TrimArray(Values, k);

    string arima101 = ARIMA101(FirstDifferences, Coefs_FD, k-1);
    string[] data = arima101.Split('\n');

    int n = data.Length;

    string[] values = data[n - 1].Split(' ');
    int m = values.Length;

    if (m == 6)
    {
        string value = values[m - 1];

        double fcst = double.Parse(value);

        int s = val.Length;
        double forecast = val[s - 1] + fcst;
    }
}
```

```

        string model = string.Format("Xt=Xt-1 + ({0})", fcst);
        msg = string.Format("ARIMA(1,1,1) model used for frecasting: \n" + model +
            "\n\nNext expected value is: " + forecast.ToString());
    }
    else
        msg = "ARIMA(1,1,1) model is not appropriate for given time series";

    return msg;
}

```



Slika 34. Poruka sa rezultatima ARIMA(1,1,1) prognoze

Slično prethodnom, ARIMA(1,2,1) se računa tako što se funkciji ARIMA(1,0,1) proslede vrednosti niza serije drugih razlika i koeficijenti autokorelacije drugih razlika umesto učitano niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija ARIMA(1,0,1) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju drugih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(1,2,1).

```

internal string ARIMA121(double[] Values, double[] SecondDifferences, double[] Coefs_SD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    string arima101 = ARIMA101(SecondDifferences, Coefs_SD, k-2);
    string[] data = arima101.Split('\n');

    int n = data.Length;

    string[] values = data[n - 1].Split(' ');
    int m = values.Length;

    if (m == 6)
    {
        string value = values[m - 1];

        double fcst = double.Parse(value);

        int s = val.Length;
        double forecast = 2 * val[s - 1] - val[s - 2] + fcst;

        string model = string.Format("Xt=2*Xt-1 - Xt-2 + ({0})", fcst);
    }
}

```

```

        msg = string.Format("ARIMA(1,2,1) used for forecasting:\n" + model + "\n\nNext  

        expected value is: " + forecast.ToString());
    }
    else
        msg = "ARIMA(1,2,1) model is not appropriate for given time series";

    return msg;
}

```



Slika 35. Poruka sa rezultatima ARIMA(1,2,1) prognoze

5.3.6. ARIMA(1,1,0) i ARIMA(1,2,0)

ARIMA(1,1,0) se računa tako što se funkciji AR(1) proslede vrednosti niza serije prvih razlika i koeficijenti autokorelacije prvih razlika umesto učitano niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija AR(1) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju prvih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(1,1,0).

```

internal string ARIMA110(double[] Values, double[] FirstDifferences, double[] Coefs_FD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    string ar1 = AR1(FirstDifferences, Coefs_FD, k-1);
    string[] data = ar1.Split('\n');

    int n = data.Length;

    string[] values = data[n - 1].Split(' ');
    int m = values.Length;

    string value = values[m - 1];

    double fcst = double.Parse(value);

    int s = val.Length;
}

```

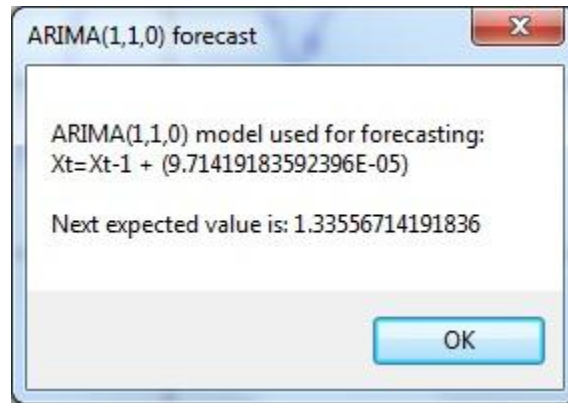
```

double forecast = val[s - 1] + fcst;

string model = string.Format("Xt=Xt-1 + ({0})", fcst);
msg = string.Format("ARIMA(1,1,0) model used for forecasting:\n" + model + "\n\nNext
expected value is: " + forecast.ToString());

return msg;
}

```



Slika 36. Poruka sa rezultatima ARIMA(1,1,0) prognoze

Analogno kao u prethodnom slučaju, ARIMA(1,2,0) se računa tako što se funkciji AR(1) proslede vrednosti niza serije drugih razlika i koeficijenti autokorelacije drugih razlika umesto učitano niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija AR(1) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju drugih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(1,2,0).

```

internal string ARIMA120(double[] Values, double[] SecondDifferences, double[] Coefs_SD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    string ar1 = AR1(SecondDifferences, Coefs_SD, k-2);
    string[] data = ar1.Split('\n');

    int n = data.Length;

    string[] values = data[n - 1].Split(' ');
    int m = values.Length;

    string value = values[m - 1];

    double fcst = double.Parse(value);

    int s = val.Length;
    double forecast = 2 * val[s - 1] - val[s - 2] + fcst;
}

```

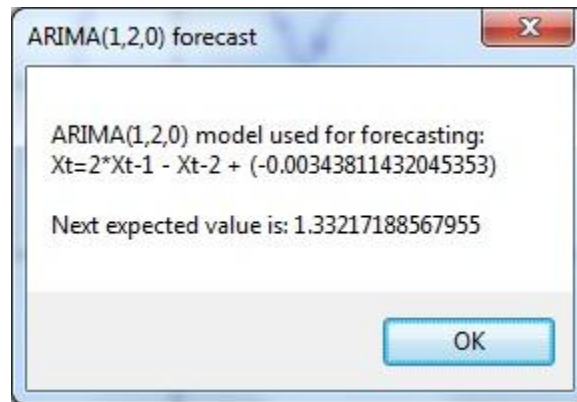


```

string model = string.Format("Xt=2*Xt-1 - Xt-2 + ({0})", fcst);
msg = string.Format("ARIMA(1,2,0) model used for forecasting:\n" + model + "\n\nNext
expected value is: " + forecast.ToString());

return msg;
}

```



Slika 37. Poruka sa rezultatima ARIMA(1,2,0) prognoze

5.3.7. ARIMA(0,1,1) i ARIMA(0,2,1)

ARIMA(0,1,1) se računa tako što se funkciji MA(1) proslede vrednosti niza serije prvih razlika i koeficijenti autokorelacije prvih razlika umesto učitano g niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija MA(1) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju prvih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(0,1,1).

```

internal string ARIMA011(double[] Values, double[] FirstDifferences, double[] Coefs_FD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    if (Math.Abs(Coefs_FD[1]) > 0.5)
        msg = string.Format("MA(1) model is not appropriate for given time series");
    else
    {
        string ma1 = MA1(FirstDifferences, Coefs_FD, k-1);
        string[] data = ma1.Split('\n');

        int n = data.Length;

        string[] values = data[n - 1].Split(' ');
    }
}

```

```

        int m = values.Length;

        string value = values[m - 1];

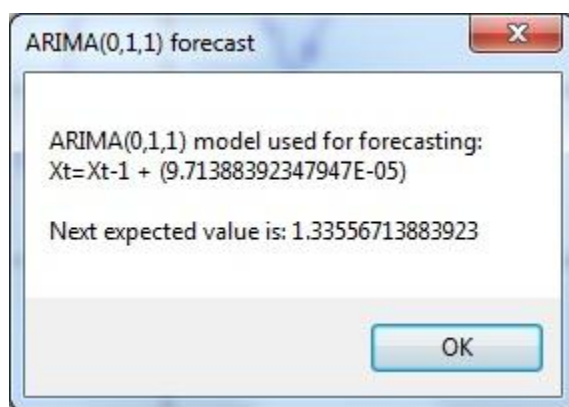
        double fcst = double.Parse(value);

        int s = val.Length;
        double forecast = val[s - 1] + fcst;

        string model = string.Format("Xt=Xt-1 + ({0})", fcst);
        msg = string.Format("ARIMA(0,1,1) model used for forecasting:\n" + model +
            "\n\nNext expected value is: " + forecast.ToString());
    }

    return msg;
}

```



Slika 38. Poruka sa rezultatima ARIMA(0,1,1) prognoze

Na sličan način, ARIMA(0,2,1) se računa tako što se funkciji MA(1) proslede vrednosti niza serije drugih razlika i koeficijenti autokorelacije drugih razlika umesto učitane niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija MA(1) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju drugih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(0,2,1).

```

internal string ARIMA021(double[] Values, double[] SecondDifferences, double[] Coefs_SD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    if (Math.Abs(Coefs_SD[1]) > 0.5)
        msg = string.Format("MA(1) model is not appropriate for given time series");
    else
    {
        string ma1 = MA1(SecondDifferences, Coefs_SD, k-2);
        string[] data = ma1.Split('\n');
    }
}

```

```

    int n = data.Length;

    string[] values = data[n - 1].Split(' ');
    int m = values.Length;

    string value = values[m - 1];

    double fcst = double.Parse(value);

    int s = val.Length;
    double forecast = 2 * val[s - 1] - val[s - 2] + fcst;

    string model = string.Format("Xt=2*Xt-1 - Xt-2 + ({0})", fcst);
    msg = string.Format("ARIMA(0,2,1) model used for forecasting:\n" + model +
        "\n\nNext expected value is: " + forecast.ToString());
}

return msg;
}

```



Slika 39. Poruka sa rezultatima ARIMA(0,2,1) prognoze

5.3.8. ARIMA(2,1,0) i ARIMA(2,2,0)

ARIMA(2,1,0) se računa tako što se funkciji AR(2) proslede vrednosti niza serije prvih razlika i koeficijenti autokorelacije prvih razlika umesto učitano g niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija AR(2) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju prvih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(2,1,0).

```

internal string ARIMA210(double[] Values, double[] FirstDifferences, double[] Coefs_FD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    string ar2 = AR2(FirstDifferences, Coefs_FD, k-1);
    string[] data = ar2.Split('\n');
}

```

```

int n = data.Length;

string[] values = data[n - 1].Split(' ');
int m = values.Length;

string value = values[m - 1];

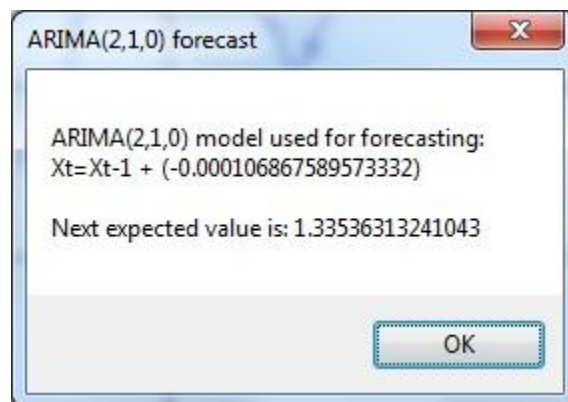
double fcst = double.Parse(value);

int s = val.Length;
double forecast = val[s - 1] + fcst;

string model = string.Format("Xt=Xt-1 + ({0})", fcst);
msg = string.Format("ARIMA(2,1,0) model used for forecasting:\n" + model + "\n\nNext  
expected value is: " + forecast.ToString());

return msg;
}

```



Slika 40. Poruka sa rezultatima ARIMA(2,1,0) prognoze

Slično tome, ARIMA(2,2,0) se računa tako što se funkciji AR(2) proslede vrednosti niza serije drugih razlika i koeficijenti autokorelacije drugih razlika umesto učitano niza i njegovih koeficijenata autokorelacije. String koji vraća funkcija AR(2) potrebno je parsirati da bi se došlo do izračunate prognoze za seriju drugih razlika koja se dalje koristi u formiranju modela, izračunavanju prognoze za učitane podatke i formiranje stringa koji vraća funkcija ARIMA(2,2,0).

```

internal string ARIMA220(double[] Values, double[] SecondDifferences, double[] Coefs_SD, int k)
{
    string msg = string.Empty;

    double[] val = TrimArray(Values, k);

    string ar2 = AR2(SecondDifferences, Coefs_SD, k-2);
    string[] data = ar2.Split('\n');

    int n = data.Length;

```

```

string[] values = data[n - 1].Split(' ');
int m = values.Length;

string value = values[m - 1];

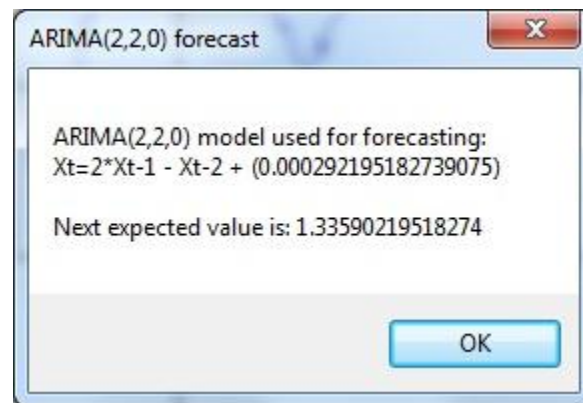
double fcst = double.Parse(value);

int s = val.Length;
double forecast = 2 * val[s - 1] - val[s - 2] + fcst;

string model = string.Format("Xt=2*Xt-1 - Xt-2 + ({0})", fcst);
msg = string.Format("ARIMA(2,2,0) model used for forecasting:\n" + model + "\n\nNext
expected value is: " + forecast.ToString());

return msg;
}

```



Slika 41. Poruka sa rezultatima ARIMA(2,2,0) prognoze

6. Primer analize tržišta. Otvaranje i zatvaranje transakcije.

Grafikoni za analizu:

1. EURUSD 1h, 15.08.2013 – 17.09.2013
2. EURUSD 24h, 01.01.2013 – 17.09.2013

Pravi se analiza za cenu zatvaranja na dan 18.09.2013 korišćenjem osnovnih alata aplikacije MetaTrader4 na prvi grafikon i primenom odgovarajućeg matematičkog modela aplikacije ImFiStBer na drugi grafikon. Ulazak u transakciju se potvrđuje primenom tehničke analize na grafikon manjeg vremenskog intervala, čiji smer trgovanja se poklapa sa prognozom koja je dobijena analizom grafikona većeg vremenskog intervala.



Slika 42. Grafikon EURUSD, 1h



Slika 43. Grafikon EURUSD, 1h, sa nacrtanom linijom trenda

Analizom linije trenda [8][9] zaključuje se sledeće:

1. Trend je rastući.
2. Potencijalno najbolji trenutak za otvaranje transakcije je kada cena dodirne liniju trenda.



Slika 44. Grafikon EURUSD, 1h, nakon primene fibonačijevog alata

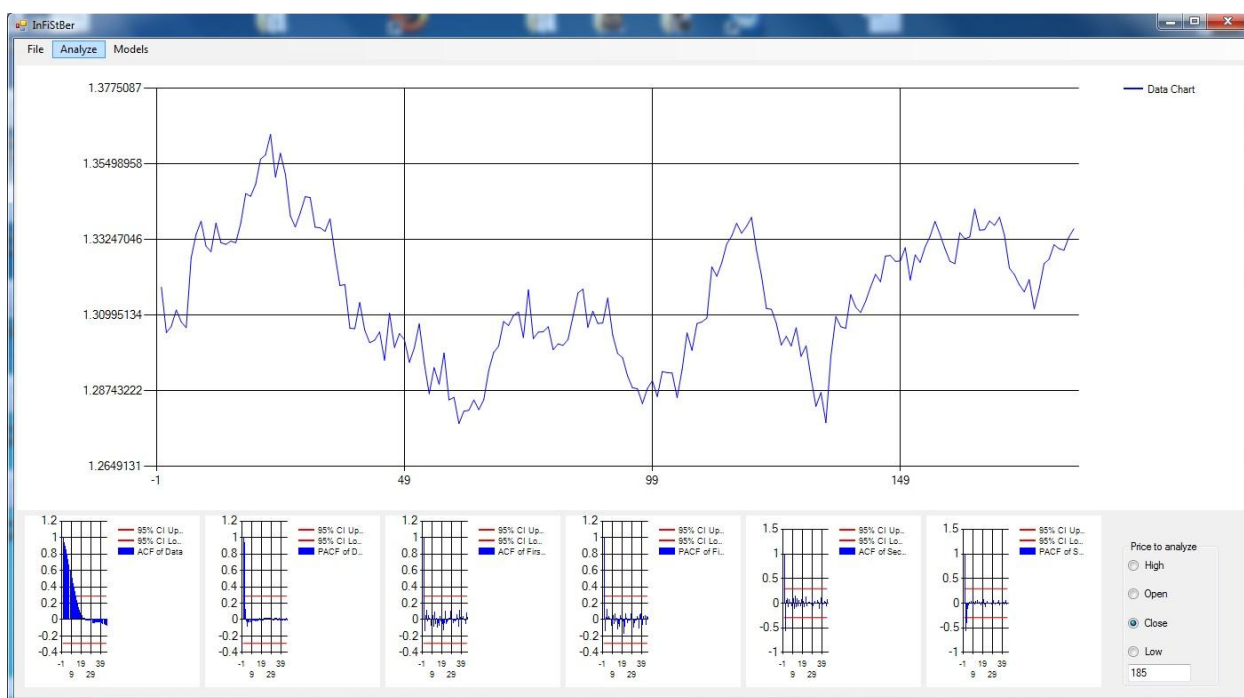
Očekivane vrednosti [8][9] primenom fibonačijeve analize:

3. 1,34390.
4. 1,35296.

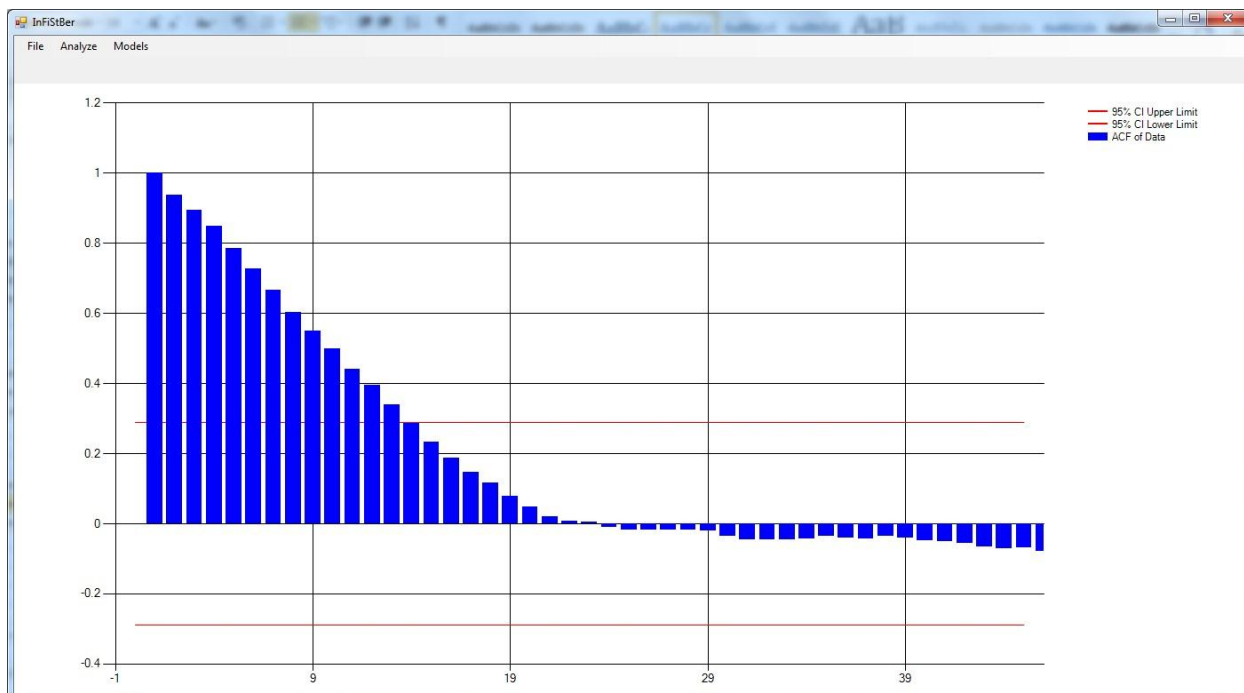


Slika 45. Grafikon dodiruje liniju trenda

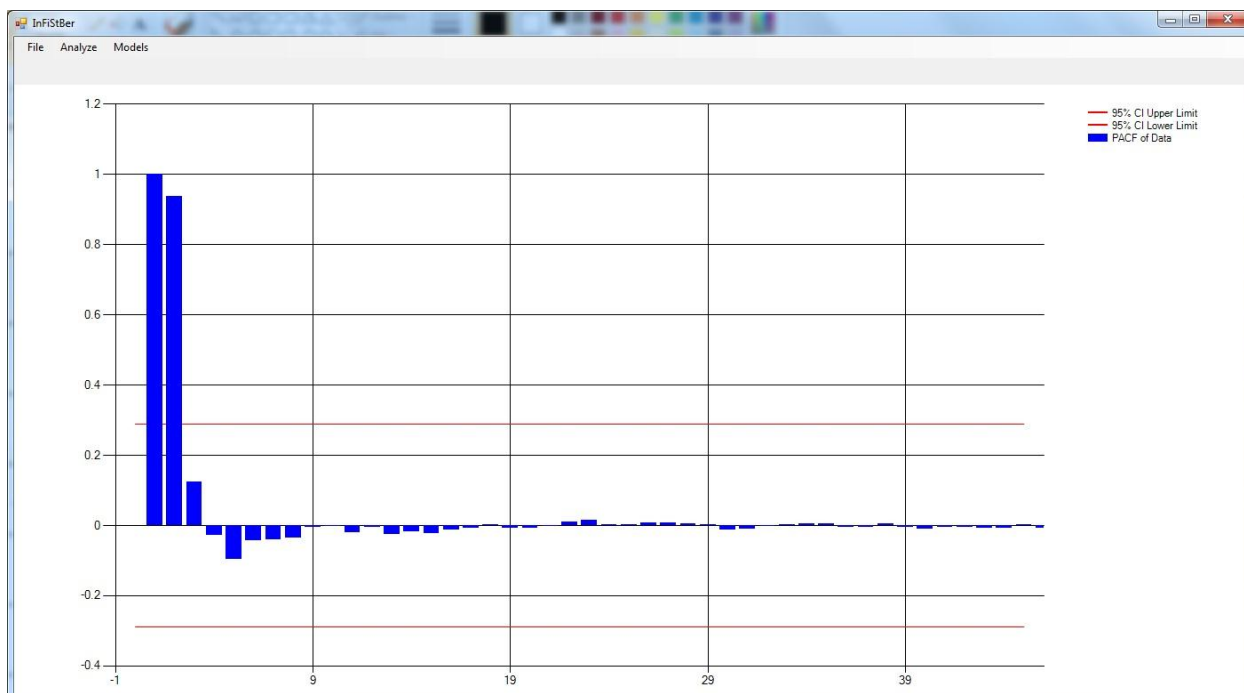
18.09.2013 u 16h grafikon dodiruje liniju trenda pri ceni od 1,3345 i to se smatra najboljim trenutkom za otvaranje transakcije na strani Bikova (eng. Bulls) jer se očekuje da će cena rasti.



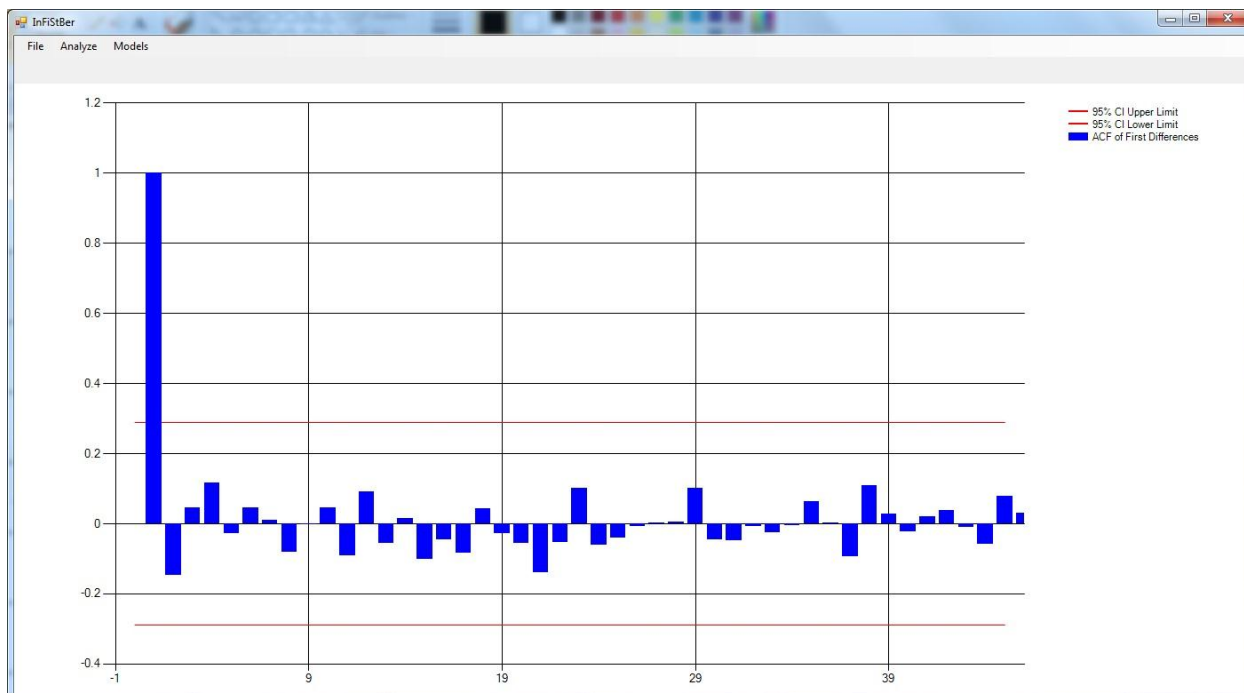
Slika 46. Grafikon EURUSD, 24h



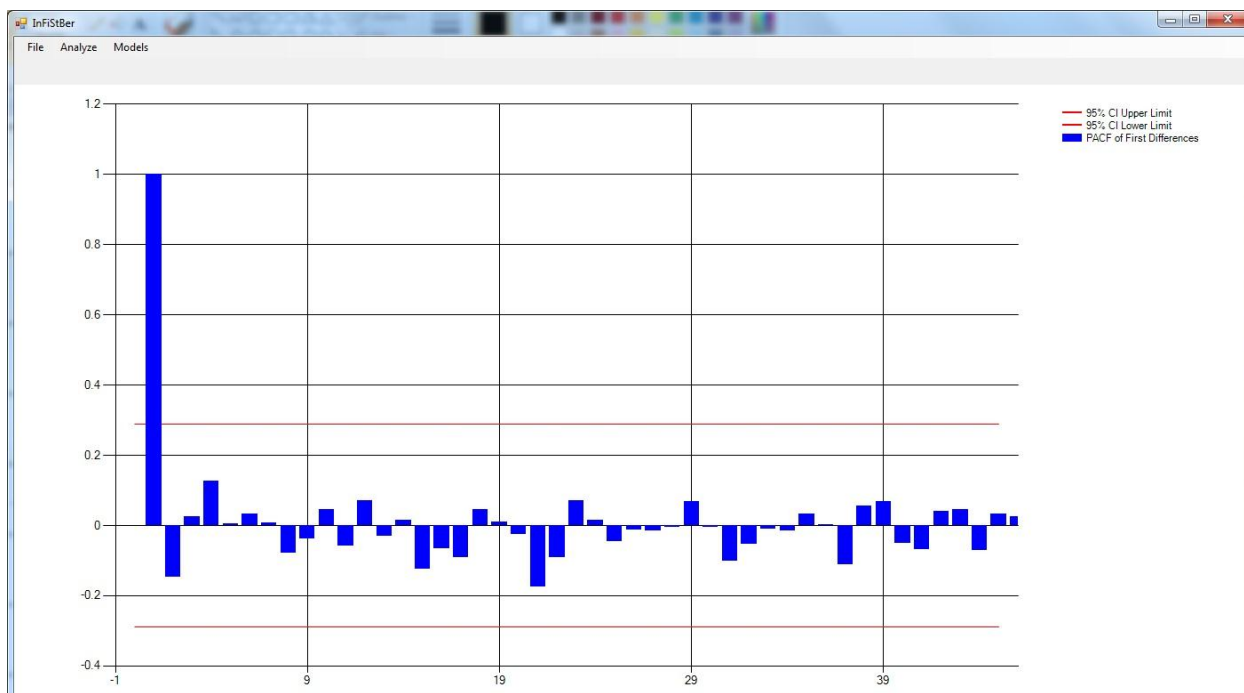
Slika 47. ACF podataka (EURUSD, 24h)



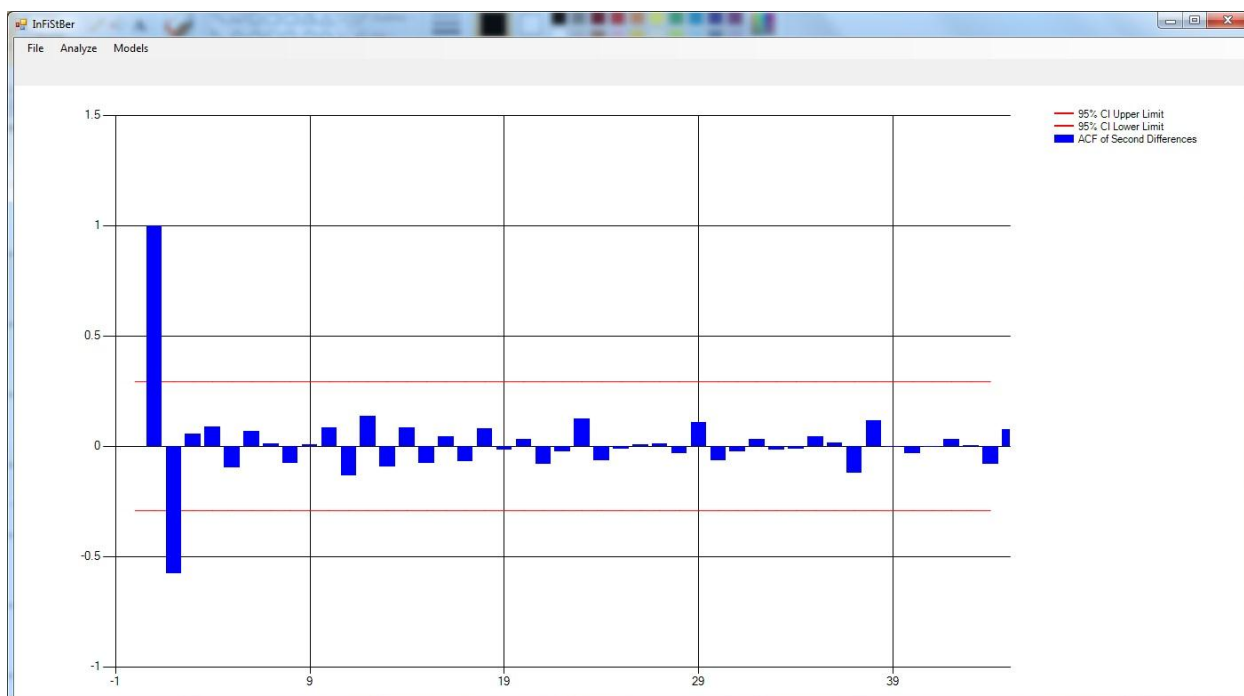
Slika 48. PACF podataka (EURUSD, 24h)



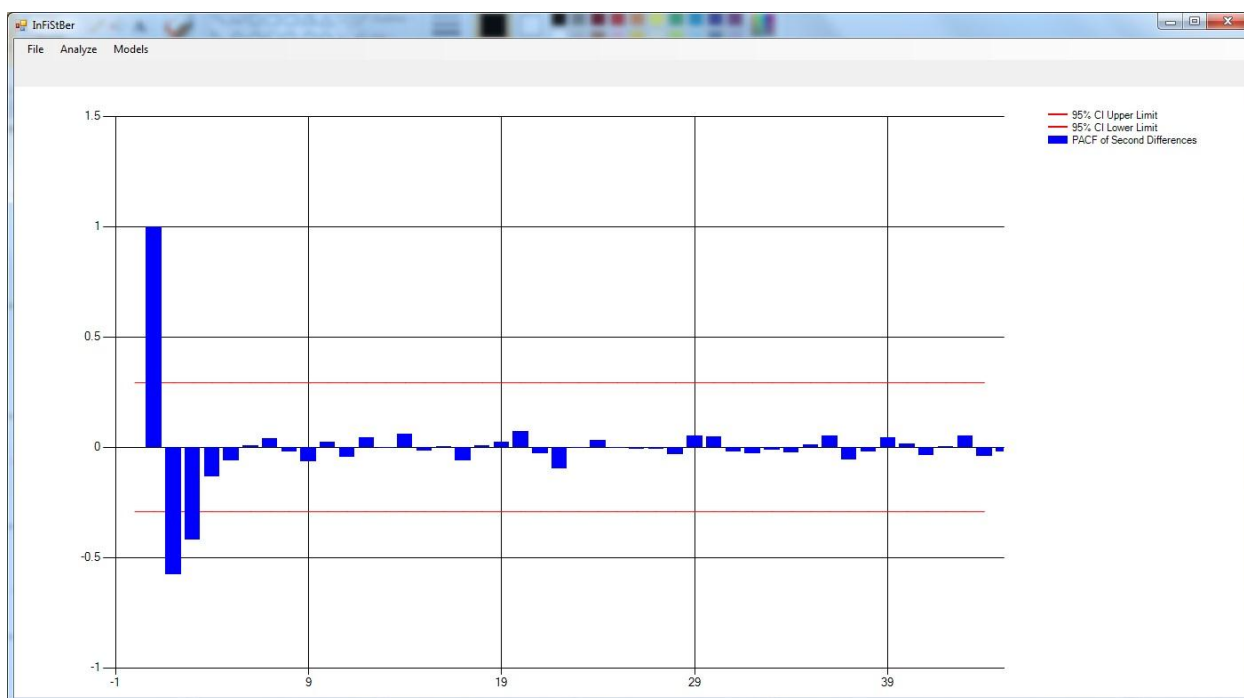
Slika 49. ACF prvih razlika (EURUSD, 24h)



Slika 50. PACF prvih razlika (EURUSD, 24h)



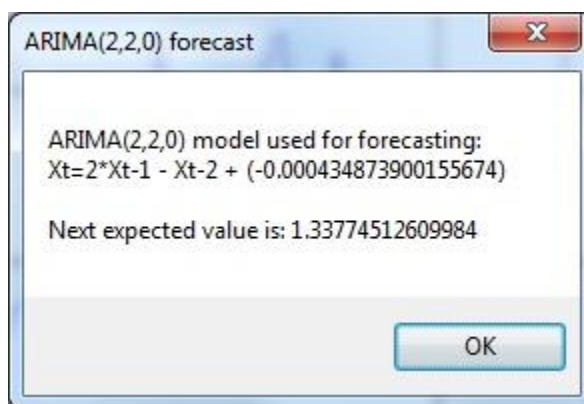
Slika 51. ACF drugih razlika (EURUSD, 24h)



Slika 52. PACF drugih razlika (EURUSD, 24h)

Analizom grafikona koeficijenata autokorelacije i parcijalne autokorelacije zaključuje se sledeće:

1. Početna serija nije stacionarna.
2. Nakon prvog diferenciranja ulaznih podataka dobija se serija koja predstavlja beli šum ($p=0$, $d=1$, $q=0$).
3. Nakon drugog diferenciranja ulaznih podataka dobija se serija za koju sledeći modeli potencijalno odgovaraju: ARIMA(1,2,0), ARIMA(2,2,0), ARIMA(0,2,1) i ARIMA (1,2,1).
4. Aplikacija modele ARIMA(0,2,1) i ARIMA (1,2,1) odbacuje jer nisu pogodni za učitane podatke.
5. Analizom grafikona koeficijenata parcijalne autokorelacije koristeći tabelu 1. (strana 23) zaključuje se da je model koji najbolje opisuje učitane podatke ARIMA(2,2,0).



Slika 53. Prognoza ARIMA(2,2,0) modela za učitane podatke

Strategija za otvaranje i zatvaranje transakcije:

1. Analiza linije trenda i fibonačijeva analiza očekuju skok cena.
2. ARIMA(2,2,0) prognoza očekuje skok cena.
3. Ukoliko grafikon dodirne liniju otvara se transakcija.
4. Prva profit zona iznosi 1,337745 dobijena ARIMA(2,2,0) prognozom.
5. Ukoliko ona brzo i lako bude probijena, očekuje se da će cene napredovati do prve fibonačijeve prognoze koja iznosi 1,34390.
6. Ukoliko i ona bude brzo i lako probijena očekuje se da će cene napredovati do druge fibonačijeve prognoze koja iznosi 1,35296.
7. Transakcija se zatvara u sledećim slučajevima:
 - a) analiza nije tačna.
 - b) nakon ostvarivanja određene profit zone.
 - c) nakon završetka radnog dana.

Grafikon dodiruje liniju trenda pri ceni od 1,3345 i dolazi do očekivanog rasta cena. Prve dve profit zone su lako bile savladane što daje sigurnost da je analiza odrađena tačno i kvalitetno. Cena zatvaranja za dan 18.09.2013 je iznosila 1,35206 što je za svega 9 Pip-a manja od druge profit zone dobijene nakon primene fibonačijeve analize. Ukupan generisan profit, od trenutka ulaska u transakciju (dodirivanje linije trenda) do trenutka izlaska (u ovom slučaju kraj radnog dana), iznosi 175 Pip-a, odnosno 1750\$ ukoliko bi

transakcija bila otvorena u iznosu od 1 Lot (100.000,00\$), koji se smatra minimalnim iznosom za poslovanje na finansijskim tržištima.



Slika 54. EURUSD, 1h, 18.09.2013, postignuta cena zatvaranja 1,35206

Zaključak

U okviru ove master teze data je implementacija finansijskih strategija koje se baziraju na ARIMA modelima za berzanska poslovanja. U radu su prikazane sve faze u ovoj aktivnosti, od samog očitavanja podataka koje dobijamo preko aplikacije MetaTrader4, do prognoze sledeće vrednosti koju daje aplikacija ImFiStBer razvijena u okviru rada na ovoj master tezi.

Kako bi korisnik mogao što lakše da donese odluku o ulasku u transakciju, pored mnogobrojnih tehničkih indikatora koje nudi MetaTrader4 i koji se koriste u svakodnevnim analizama finansijskih tržišta u većini brokerskih kuća, korisnik aplikaciju ImFiStBer može da koristi za dublju analizu tržišta.

U našoj zemlji poslovanje na svetskom finansijskom tržištu nije zastupljeno u onoj meri kao što je to slučaj u drugim zemljama. Poslednjih godina došlo je do povećanja broja brokerskih kuća koje nude mogućnost trgovanja na svetskom tržištu, ali još uvek samo poslovanje nije zakonski regulisano.

Iz dosadašnjeg iskustva u poslovanju na finansijskom tržištu, može se zaključiti da aplikacija olakšava donošenje odluke za ulazak u transakciju. Uz prisustvo drugih tehničkih indikatora kao i same aplikacije, analiza finansijskih tržišta postaje preciznija i olakšana.

Uz male modifikacije aplikacija ImFiStBer može se koristiti za analizu različitih vremenskih serija npr. serija merenja temperature u nekom hemijskom procesu, godišnji broj sunčevih pega i mnoge druge vremenske serije.

Napomena

Kompletan izvorni kod potreban za kreiranje ImFiStBer aplikacije opisane u ovom radu nalazi se na CD-u koji je priložen uz master tezu.

Literatura

- [1] Zorica Mladenović, Aleksandra Nikolić (2008), *Analiza vremenskih serija: Primeri iz srpske privrede*
- [2] George Box, Gwilym Jenkins, Gregory Reinsel (2008), *Time Series Analysis*
- [3] Søren Bisgaard, Murat Kulahci (2011), *Time Series Analysis and Forecasting by Example*
- [4] Alen Belullo (2011), *Uvod u ekonometriju: Sveučilišni udžbenik*
- [5] John Cochrane (1997), *Time Series for Macroeconomics and Finance*
- [6] Brian Borchers (2001), *The Partial Autocorrelation Function*
- [7] Derick Boyd (2007), *Autocorrelation, Auto-regression, Serial Corelation*
- [8] John L. Person (2007), *FOREX Conquered: High Probability Systems and Strategies for Active Traders*
- [9] Ed Ponsi (2007), *FOREX Patterns and Probabilities: Trading Strategies for Trending and Range-Bound Markets*
- [10] John L. Person (2004), *A Complete Guide to Technical Trading Tactics*