



Akademia Górniczo – Hutnicza
im. Stanisława Staszica w Krakowie
Wydział Elektrotechniki, Informatyki, Automatyki i Elektroniki

Rozprawa Doktorska

Zastosowanie sieci neuronowych do generowania
strategii decyzyjnych (inwestycyjnych)
na przykładzie Giełdy Papierów Wartościowych.

mgr Radomir Domaradzki

Promotor:

prof. dr hab. inż. Ryszard Tadeusiewicz

Kraków 2007

Składam serdeczne podziękowania
Panu Profesorowi Ryszardowi Tadeusiewiczowi
Za poświęcony czas, życzliwość oraz wszelką pomoc,
która przyczyniła się do powstania tej pracy.

Spis treści

1. Wstęp	5
1.1. Koncepcje badawcze i teza rozprawy	7
2. Wprowadzenie	10
2.1. Rys historyczny rozwoju sieci neuronowych	11
2.2. Inspiracje biologiczne	15
2.3. Podstawowy model sztucznego neuronu	17
2.4. Budowa i zasada funkcjonowania sztucznych sieci neuronowych	20
2.5. Metody uczenia sztucznych sieci neuronowych	22
2.5.1. Reguła Delta	23
2.5.2. Reguła Hebba	25
3. Modele prognostyczne	28
3.1. Dobór zmiennych wejściowych	28
3.2. Charakterystyka wektora wejściowego	31
3.3. Optymalizacja wektora zmiennych wejściowych	34
3.3.1. Wyznaczenie optymalnej sieci dla 56 zmiennych wejściowych	35
3.3.2. Analiza głównych składowych (Principal Components Analysis)	40
3.3.3. Wyznaczenie optymalnego wektora wejściowego przy pomocy analizy wrażliwości	43
3.3.4. Wyznaczenie optymalnego wektora wejściowego przy pomocy Automatycznego Projektanta Sieci	46
3.3.5. Dobór zmiennych wejściowych przy pomocy algorytmu genetycznego	48
3.3.6. Wyznaczenie optymalnego wektora wejściowego przy pomocy nieliniowej kompresji danych	51
3.3.7. Wyznaczenie optymalnego wektora wejściowego przy pomocy selekcji krokowej wstecznej i postępującej	53
3.4. Podsumowanie	55
4. Prognozowanie wartości indeksu WIG20	56
4.1. Prognozowanie wartości indeksu WIG20 na podstawie jego wartości z poprzednich okresów – typowa neuronowa analiza szeregów czasowych	56

4.1.1. Dobór zmiennych wejściowych	57
4.1.2. Model prognostyczny	57
4.2. Prosta strategia decyzyjna w oparciu o prognozę wartości indeksu WIG20 z wykorzystaniem wielu zmiennych wejściowych	60
4.3. Prognozowanie wartości zamknięcia z przesunięciem w czasie	63
4.4. Podsumowanie	66
5. Neuronowe modele decyzyjne	67
5.1. Porównanie sieci neuronowych typu MLP oraz RBF	67
5.2. Krótkoterminowe neuronowe strategie decyzyjne	70
5.3. Neuronowe strategie decyzyjne oparte o prognozowany trend	75
5.4. Podsumowanie	80
6. Wykorzystanie samoorganizujących sieci neuronowych Kohonena do budowy modeli decyzyjnych	81
6.1. Sieć Kohonena – krótka charakterystyka	81
6.2. Sieć neuronowa Kohonena jako narzędzie wstępnego grupowania wektorów wartości wejściowych	85
6.3. Podsumowanie	92
7. Zakończenie	93
Dodatek A (Metody analizy i prognozowania finansowych szeregów czasowych)	96
Dodatek B (Istota giełdowego kontraktu terminowego, należącego do instrumentów pochodnych notowanych na Giełdzie Papierów Wartościowych w Warszawie)	102
Literatura	106

1. Wstęp

Przewidywanie przyszłych zdarzeń ma ogromne znaczenie w procesach sterowania i zarządzania, zwłaszcza gdy chodzi o zarządzanie procesami gospodarczymi. Ceny akcji na giełdzie, kursy walut, ceny obligacji czy metali szlachetnych, wartość inflacji czy wielkość PKB, to tylko niektóre z wielkości ekonomicznych, które mają kolosalne znaczenie w prowadzeniu działalności gospodarczej. Możliwość trafnego prognozowania przyszłych wartości tych wielkości ekonomicznych lub - co często ważniejsze - umiejętność w miarę dokładnego odgadywania przyszłych **kierunków zmian** cen akcji, walut czy surowców, może stanowić dla firmy istotne źródło sukcesów gospodarczych. Może być to także decydujący czynnik, pozwalający jednym firmom osiągnąć trwałą przewagę konkurencyjną i utrzymać się na rynku w sytuacji, gdy wiele innych firm jest zagrożonych bankructwem. Nic dziwnego, że w opisanych warunkach wszelkie narzędzia informatyczne, pozwalające uzyskiwać dobre prognozy ekonomiczne, były i są uważane za bardzo cenne.

Fakt, że pewne wyniki naukowe są poszukiwane i cenne skutkuje natychmiast tym, że bardzo wielu badaczy na całym świecie poświęca swój talent i pracę w tym celu, aby właśnie te cenione i poszukiwane wyniki naukowe uzyskać. W **dodatku A** do tej pracy przedstawiono krótki przegląd różnych metod prognozowania procesów ekonomicznych, które można znaleźć we współczesnej literaturze. Jakkolwiek przegląd ten jest zapewne niepełny i niekompletny, to jednak pokazuje on, że na temat sposobów przewidywania przyszłych wartości takich czy innych ekonomicznych szeregów czasowych napisano już bardzo wiele wartościowych prac, więc sam ten temat, traktowany autonomicznie, utracił już wiele ze swojej pierwotnej atrakcyjności naukowej.

Jednak, jak zostanie pokazane w dalszej części tej pracy, sama tylko predykcja procesów gospodarczych nie jest najwygodniejszą ani najkorzystniejszą formą komputerowego wspomaganie ekonomicznych procesów decyzyjnych. Wynika to z faktu, że w ostatnim czasie mamy do czynienia (w procesach ekonomicznych oraz w zarządzaniu nimi) także i bez narzędzi prognostycznych z prawdziwym zalewem informacji pochodzących z różnych źródeł. Decydent, który dawniej narzekał na brak informacji, obecnie zaczyna cierpieć z powodu ich nadmiaru, a co się z tym wiąże, obok problemu dotyczącego tego, skąd zdobyć potrzebne informacje (między innymi prognozy gospodarcze) powstaje problem ich szybkiego i efektywnego przetwarzania oraz selekcji. Dlatego w niniejszej pracy, poświęconej w istocie zagadnieniom prognozowania zjawisk i procesów gospodarczych,

spróbujemy zerwać z tradycyjnym sposobem formułowania i wykorzystywania prognozy. Zamiast odrębnie opracowywać model, który dostarczy prognozowanych wartości rozważanych parametrów, a potem odrębnie budować system komputerowy, który pomoże decydentowi w podjęciu właściwej decyzji (na przykład dotyczącej inwestowania) w oparciu o dane wynikające z prognozy (oraz ewentualnie także inne dane) – podejmiemy próbę budowy systemu doradczego, który będzie od razu tworzył **radę inwestycyjną** albo wręcz generował sygnały kupna-sprzedaży. Postaramy się wykazać, że taki system będzie miał przewagę nad systemem, w którym komputer najpierw buduje model prognostyczny, z pomocą którego użytkownik będzie analizował rozważane zjawisko ekonomiczne, a potem ten sam użytkownik będzie angażował drugi system, który pozwala zorientować się w znaczeniu wyników prognoz i który będzie pełnił rolę doradczą w procesie podejmowania decyzji.

Wskazane wyżej kompleksowe rozwiązania problemu prognostyczno-decyzyjnego ma bezsporne zalety zwłaszcza wtedy, gdy podczas podejmowania decyzji trzeba brać pod uwagę równocześnie wiele okoliczności i wiele czynników warunkujących sukces lub niepowodzenie całego działania. W takich złożonych przypadkach ogrom danych koniecznych do uwzględnienia może być wręcz niewyobrażalny i niejednokrotnie niemożliwy do objęcia przez badaczy, menadżerów czy potencjalnych inwestorów. Odwołajmy się do konkretnego przykładu podejmowania decyzji związanych z inwestycjami na rynkach finansowych.

Zarówno giełdy akcji jak i giełdy towarowe funkcjonują w kilkudziesięciu krajach świata. Na każdej z nich notowanych jest po kilkaset, a nawet kilka tysięcy walorów. Ponadto większość krajów ma różne waluty narodowe, różny poziom inflacji, z każdym krajem wiąże się różne ryzyko inwestycyjne i każdy z nich ma inną politykę gospodarczą, np. politykę stóp procentowych. Daje to kolejne tysiące wartości, które codziennie mogą być okazją do zarobienia, bądź stracenia pieniędzy. Wiadomo, że nikt nie jest w stanie przeanalizować ani nawet zarejestrować wszystkich tych sytuacji, zwłaszcza jeśli należą one do różnych klas zdarzeń. Jednak dobry analityk powinien uwzględnić maksymalnie dużo informacji mających wpływ na dany sektor, aby móc z nich wyciągać jak najtrafniejsze wnioski.

Jak to osiągnąć?

Tradycyjne techniki przetwarzania informacji nie zawsze dają oczekiwany efekt w rozsądnym czasie. Ponadto w szybko zmieniających się warunkach działania wielu przedsiębiorstw i związanych z nimi systemów informatycznych - pojawia się konieczność szybkiego dostosowywania istniejących algorytmów do nowych warunków, a to nie jest rzeczą ani łatwą, ani tanią.

Często także nie istnieją ani teoretyczne ani praktyczne rozwiązania dla przedstawionych powyżej problemów. Tymczasem rozwiązanie przynajmniej niektórych z nich może być znalezione, pod warunkiem, że w obszar rozwiązań dopuszczalnych włączone zostaną metody niestandardowe (jak chociażby proponowane w tej pracy sieci neuronowe), co na pewno przyniesie korzyść dla gospodarki, techniki i dla indywidualnych podmiotów.

1.1. Koncepcje badawcze i teza rozprawy

Sieci neuronowe ze względu na swoją specyfikę są wręcz nieocenionym narzędziem do modelowania wszelkiego rodzaju zjawisk, w których niemożliwe jest prześledzenie zachodzących w nich związków o charakterze przyczynowo-skutkowym. Takie zjawiska, opisywane w cybernetyce modelem „czarnej skrzynki” bardzo dobrze nadają się do opisu wielu interesujących procesów ekonomicznych. W szczególności modele tego rodzaju nadają się do prognozowania cen walut, akcji, modelowania strategii inwestycyjnych, itp. W tym obszarze nie ma zazwyczaj znanych modeli ekonometrycznych, które dobrze opisywałyby powyższe zagadnienia. Ale sieci neuronowe ze swoją zdolnością do **odkrywania** zależności między danymi wejściowymi a danymi wyjściowymi poprzez proces **nauki** na bazie przykładowych zadań, dla których znane są poprawne rozwiązania - wypełniają tę lukę. Jednak sama umiejętność uczenia się nie jest kluczem do sukcesu. Bardzo ważną cechą sieci neuronowych jest także ich zdolność do **generalizacji** zdobytej wiedzy, co oznacza możliwość przeniesienia odkrytego w toku procesu uczenia zbioru reguł na nowe przypadki, które nie występowały podczas procesu uczenia, ale które oczywiście muszą podlegać tym samym regułom, jakie determinowały przypadki uczące. Innymi słowy przy używaniu modeli neuronowych musimy mieć do czynienia z sytuacją w której jakieś reguły **istnieją**, tylko są nieznane. Gdyby wspomnianych reguł nie było, na przykład gdyby rozważane zjawisko podlegało wyłącznie losowym fluktuacjom, to żaden model – neuronowy czy jakiegokolwiek inny – nie byłby w stanie tego zjawiska prognozować.

Odwołując się do sieci neuronowych jako do narzędzia rozwiązywania wybranych problemów ekonomicznych musimy mieć świadomość ich ograniczeń, na przykład tego, że wyniki obliczeń neuronowych są zwykle raczej jakościowe, niż ilościowe. Dokładniej można to wyrazić następującą regułą: Sieci neuronowe, jak każde narzędzie obliczeniowe, w następstwie przeprowadzonych procesów przetwarzania wejściowych danych dostarczają na wyjściu pewnych wyników, które zawsze mają postać liczb. Jednak jeśli te liczby traktuje się jako wynik rozwiązania określonego problemu matematycznego i w rezultacie wymaga się od nich wysokiej dokładności - to z reguły okazują się, że neuronowy model tym wymaganiom nie może sprostać, ponieważ ze sposobu przetwarzania danych zachodzącego w sieci wynika, że rozwiązania wypracowane przez sieć są zawsze w większym lub mniejszym stopniu **przybliżone**. Fakt ten mógłby być bardzo frustrujący (w końcu niedokładnie liczące urządzenie informatyczne wydaje się na pierwszy rzut oka znacznie gorsze od takiego, które wszystko oblicza bardzo dokładnie!), gdyby nie fakt, że w wielu zastosowaniach informatyki dokładny wynik numeryczny nie jest bynajmniej tym, na czym nam najbardziej zależy!

Skupiając uwagę na głównym wątku merytorycznym tej pracy można powiedzieć, że przy podejmowaniu decyzji ekonomicznych, na przykład przy inwestowaniu na Giełdzie Papierów Wartościowych, wcale nie musimy znać **dokładnej** wartości prognozowanych cen odpowiednich akcji lub innych papierów wartościowych. Tym, co nas w istocie interesuje, jest wspomaganie procesu podejmowania decyzji – czy określone walory należy (w świetle prognozowanych zmian cen) kupić, sprzedać, czy przechować?

W oparciu o powyższe stwierdzenie można zdefiniować następującą tezę:

Doradczy system inwestycyjny, oparty na neuronowym modelu decyzyjnym, może przynosić znacząco lepsze wyniki na Giełdzie Papierów Wartościowych niż inwestowanie w oparciu o prognozy cen Papierów Wartościowych uzyskane także przy pomocy sieci neuronowych.

Drugą tezę, związaną z wyżej sformułowaną, którą będziemy chcieli udowodnić w ramach tej rozprawy, będzie teza szczegółowa:

Samoorganizujące sieci neuronowe Kohonena mogą być wykorzystywane do budowy modeli decyzyjnych, a efektywność takich modeli może być większa niż prognozowanie wartości kursu akcji przy pomocy sieci neuronowych typu MLP.

Realizując tematykę badawczą zdefiniowaną przez sformułowaną powyżej tezę, w pracy przedstawione zostaną dwie odrębne grupy modeli. Pierwsza grupa to tradycyjne modele prognostyczne, których zadaniem będzie prognozowanie zmiennej wyjściowej (wartości indeksu) o jeden lub dwa okresy naprzód. W tej części oprócz poszukiwania najlepszej struktury sieci, zaprezentowane zostaną również różne metody optymalizacji wektora zmiennych wejściowych, w taki sposób, aby do modelu wprowadzać (dobrać) tylko te zmienne, które polepszają jego właściwości prognostyczne. Wykorzystanie różnych metod dostępnych w pakiecie Statistica do doboru zmiennych wejściowych może polepszyć działanie modelu prognostycznego poprzez odrzucenie tych zmiennych, które wprowadzają nadmierny szum w stosunku do ich realnych wartości prognostycznych.

Druga grupa modeli to neuronowe modele decyzyjne, których zadaniem będzie nie stricte **prognozowanie** przyszłych wartości indeksu, ale raczej przewidywanie kierunku zmian czy występującego trendu. W tym celu wykorzystane zostanie kilka rodzajów sieci neuronowych. Zadaniem tych modeli będzie wspomaganie decyzji inwestycyjnych poprzez pokazanie inwestorowi jaką pozycję (*kupić, sprzedać, utrzymywać papiery wartościowe*) powinien zająć.

Poprzez porównanie opisanych dwóch klas modeli spróbujemy wykazać, że rozwiązanie zmierzające wprost do formułowania przez neuronowy model sugestii decyzji inwestycyjnych (lub gotowych sygnałów kupna-sprzedaży) **ma przewagę** nad podejściem opartym na odrębnym budowaniu prognoz i osobnym wspomaganiu procesów informacyjnych zmierzających do decyzji – nie wspominając o tym, że pierwsze, proponowane w tej pracy podejście, jest o wiele wygodniejsze dla użytkowników.

2. Wprowadzenie

Funkcjonowanie mózgu fascynowało ludzi od wielu pokoleń. Już w XVIII wieku próbowano konstruować urządzenia do gry w szachy, które mogłyby wykazać się samodzielną umiejętnością analizowania sytuacji i podejmowania decyzji. Jednak gra w szachy ma stosunkowo łatwo algorytmizowalne zasady, a wśród zjawisk ekonomicznych wiele jest takich, które mają tylko częściowo określone lub zupełnie nieokreślone reguły działania. Do częściowo określonych zjawisk należy na przykład rynek energii elektrycznej, gdzie wiadomo, że zachodzą zależności pomiędzy zużyciem energii, a upływem czasu (wyróżnia się cykl dzienny, tygodniowy i roczny). W przypadku rynków finansowych sytuacja wygląda zupełnie inaczej. Nie występują tam żadne jawne zależności pomiędzy na przykład ruchami cen na giełdzie. W takim przypadku, gdzie brak jest jawnych związków przyczynowo-skutkowych, należy zastosować narzędzia, które potrafią **odgadnąć** takie zależności. Chodzi oczywiście o „odgadywanie” oparte na naukowych zasadach, przy wykorzystaniu nowoczesnych narzędzi informatycznych. Jednym z takich narzędzi, któremu poświęcona jest niniejsza praca są sieci neuronowe. Posiadają one unikalne cechy takie jak zdolność uczenia się i generalizacji zdobytej wiedzy. W efekcie dostajemy system, mający umiejętność wyciągania wniosków nawet na podstawie niekompletnych bądź zaszumionych danych, a przyswajanie wiedzy przez sieci neuronowe następuje w procesie uczenia, który może odbywać się w sposób ciągły.

W teorii sieci neuronowe to algorytmy i struktury przetwarzania informacji bazujące na neurofizjologii. W praktyce sieci neuronowe to programy komputerowe, które symulują działanie mózgu, ale tak dalece upraszczają wiedzę neurocybernetyczną, że powstające narzędzie cechuje znaczna użyteczność praktyczna. Istnieje wiele symulatorów sieci neuronowych, np. programy BrainMaker czy Statistica Neural Network firmy Statsoft. Jednak sam symulator może być porównywalny do umysłu noworodka, nic nie umie i wszystkiego go trzeba nauczyć. Różnica jest taka, że dziecko poznaje jednocześnie wiele dziedzin, a niektóre reguły są mu podawane w sposób jawny. Z symulatorem sieci neuronowych jest inaczej. Musi się on nauczyć rozwiązania tylko jednego problemu, a prezentowanymi mu przykładami uczącymi są szczególne przypadki zjawisk ekonomicznych zaistniałych w przeszłości. Na podstawie zbioru tych przypadków sieć neuronowa **odgaduje** tą regułę, która rządzi danym zjawiskiem. Reguła ta ma najczęściej formę funkcji nieliniowej o wielu wejściach, która najlepiej określa zależności między wejściem i wyjściem. O sukcesie modelu

neuronowego w dużej mierze decyduje człowiek, który trenuje sieć neuronową i dobiera jej architekturę. To on musi wybrać zmienne wejściowe, ich liczbę, strukturę sieci a także algorytm jej uczenia. Co prawda istnieją narzędzia wspomagające podejmowanie takich decyzji, ale ostatecznie zdanie należy zawsze do człowieka.

Twórcy SF od dawna fascynowali się możliwościami oferowanymi w nieokreślonej przyszłości przez superkomputery. Jednak w większości przypadków komputery nie zajmowały się w sposób jawny wykorzystaniem swojej mocy obliczeniowej, lecz drzemała w nich moc, pozwalająca na analizowanie cudzych zachowań i podejmowanie decyzji adekwatnych do tych zachowań. Właśnie takie narzędzie byłoby idealne dla osób analizujących zjawiska ekonomiczne. Komputer obserwuje zachowanie rynku i podejmuje obiektywne, trafne i pozbawione emocji decyzje ekonomiczne – kupuje bądź sprzedaje akcje, obligacje, surowce bez udziału lub z niewielkim udziałem człowieka. Sytuacja taka zdawać by się mogło jest futurystyczna albo wręcz nierealna, gdyż komputery nie są w stanie bezbłędnie prognozować przyszłych zmian cen na giełdzie, czy tendencji w gospodarce. Nie oznacza to jednak, że dzisiejsza sytuacja analityków jest taka sama jak przed laty. Istnieje bowiem wiele narzędzi w postaci oprogramowania, które skutecznie mogą pomagać w analizie zjawisk ekonomicznych. Można by się nawet pokusić o budowę systemu, który samoczynnie podejmowałby decyzje inwestycyjne. Niejedna przeprowadzona symulacja na danych historycznych pokazała, że taki poprawnie zbudowany model dobrze radziłby sobie z prognozowaniem zmian na rynku i mógłby przynosić zyski. Zachodzi więc potrzeba przejścia od działań symulowanych i pozorowanych – do praktyki.

2.1. Rys historyczny rozwoju sieci neuronowych

Sieci neuronowe od czasu swych narodzin w latach czterdziestych przeżywały okresy wzlotów i upadków. Od początkowej fascynacji ich możliwościami, poprzez znaczny spadek zainteresowania, czy wręcz zapomnienia po słynnej książce Mińskiego i Paperta¹, aż po renesans w latach osiemdziesiątych i dziewięćdziesiątych.

¹ Minsky M., Papert S., „*Perceptrons*”, MIT Press, Cambridge 1969 – Ta publikacja zahamowała rozwój sieci neuronowych na prawie 15 lat, gdyż zawierała formalny dowód na to, że sieci neuronowe mają bardzo ograniczony zakres zastosowań. Dotyczyło to wprawdzie wyłącznie sieci jednowarstwowych, jednak mało kto zwrócił uwagę na tę okoliczność, a ponadto w tamtych czasach nie były jeszcze znane efektywne metody uczenia nieliniowych sieci wielowarstwowych.

Inspiracją dla tej dziedziny nauki, modelem, do którego, początkowo bezpośrednio, a potem pośrednio odwoływali się badacze sieci neuronowych - jest mózg. Ten najbardziej skomplikowany i tajemniczy z ludzkich organów stanowi dla nas ciągle zagadkę. Dzisiejsze, tak szerokie i powszechne zainteresowanie sieciami neuronowymi zarówno wśród inżynierów, przedstawicieli nauk ścisłych - matematyki i fizyki oraz biologów czy neurofizjologów, wynika przede wszystkim z poszukiwań sposobów budowy bardziej efektywnych i bardziej niezawodnych urządzeń do przetwarzania informacji. Okazuje się, że mimo ponad pół wieku doskonalenia komputerów - układ nerwowy jest tutaj wciąż niedoścignutym wzorem.

Z punktu widzenia dzisiejszej informatyki ważne jest nie tyle całościowe modelowanie mózgu, ile możliwość wykorzystania sposobów, jakimi on przetwarza informacje. Ciekawy z punktu widzenia informatyki jest także fakt, że dzięki tym modelom fragmentarycznie jedynie wykorzystującym wiedzę neurofizjologiczną, można rozwiązywać zadania, z którymi z trudem radzą sobie inne techniki obliczeniowe. Sieci neuronowe bowiem, mogą być stosowane wszędzie tam, gdzie pojawiają się problemy związane z przetwarzaniem i analizą danych, z ich predykcją, klasyfikacją czy sterowaniem. Można pokusić się nawet o stwierdzenie, że łatwiej wymienić problemy, w których sieci neuronowe się nie sprawdzają, niż te, w których z powodzeniem można je zastosować (i w których zapewne były już stosowane), bo tych ostatnich jest bardzo dużo i stale przybywają nowe.

Głównym czynnikiem przemawiającym za praktycznym stosowaniem sieci neuronowych jest ich zdolność do uogólniania zdobytej wiedzy, która daje im jak gdyby pewną dozę inteligencji. Ciekawym i równie ważnym czynnikiem jest to, że sieci neuronowe są wyposażone w swoje wewnętrzne algorytmy przetwarzania informacji, które umożliwiają im rozwiązywanie nawet gatunkowo różnych zadań. Sposób w jaki sieć neuronowa zyskuje wiedzę o zadanym problemie, polega na nauce na podstawie znanych poprawnych przykładów (zwanych wzorcami uczącymi) rozwiązania danego problemu, lub prościej: na bazie obserwacji prezentowanej jej wiedzy. Celem nauki jest sposób opisanie wewnętrznych korelacji, zachodzących pomiędzy wzorcami uczącymi. Na tej podstawie nauczona sieć neuronowa potrafi odpowiadać na pytania zarówno z zakresu wzorców uczących jak i dla zadań nie identycznych z przykładami wykorzystywanymi do uczenia, ale podlegających temu samemu schematowi logicznemu.

Mózg człowieka ciągle jest najpotężniejszym z istniejących obecnie urządzeń liczących do celów przetwarzania informacji w czasie rzeczywistym. Fascynacje mózgiem, jego własnościami (odpornością na uszkodzenia, równoległym przetwarzaniem itp.) już

w latach 40-tych zaowocowały pracami, których fundamentalne znaczenie odczuwamy jeszcze dzisiaj.

Choć niewielu z nas potrafi przemnożyć w pamięci dwie liczby dwucyfrowe, co bez problemu robi najprostsz kalkulator, każdy z nas bezbłędnie rozpoznaje różne obrazy (na przykład twarze znajomych osób), z czym komputery mają jeszcze olbrzymie problemy. W zagadnieniach wymagających heurystycznego przetwarzania informacji mózg może pokonać nawet najszybszy superkomputer, pomimo że w tym procesie neurony jako jednostki przetwarzające są o wiele rzędów wielkości (przynajmniej **milion razy!**) wolniejsze od swoich elektronicznych czy optoelektronicznych odpowiedników.

Sieci neuronowe odziedziczyły po swoim pierwowzorze kilka cech odróżniających ich działanie od innych technicznych systemów przetwarzania danych, a upodabniających je do biologicznego mózgu. I choć nie mają one monopolu na naśladowanie mózgu, to właśnie najczęściej sieci neuronowe odczytują ręczne pismo, prognozują kursy akcji na giełdzie, rozpoznają mowę, obrazy czy twarze.

Obecnie sieci neuronowe są już powszechnie znanym i cenionym narzędziem informatycznym, znajdującym coraz więcej zastosowań w różnych dziedzinach, w tym także w ekonomii. Żeby jednak kompetentnie mówić o współczesnym stanie wiedzy (i praktyki) sieci neuronowych dobrze jest przyrzeć się, z czego się ta technika wywodzi i jakie były jej początki.

W początkowym okresie, gdy nie było formalnych podstaw nauki sztucznych sieci neuronowych, zainteresowanie badaczy mocno związanych z dziedziną neurofizjologii czy bioniki skierowane były na opis ogólnych mechanizmów działania mózgu oraz na modelowanie pojedynczych komórek układu nerwowego. Samodzielnie dziedzina sztucznych sieci neuronowych zaistniała dopiero wraz z wydaniem historycznej pracy [54], w której po raz pierwszy przedstawiono matematyczny opis komórki nerwowej. Kolejnym kanonem, wyznaczającym na długi czas rozwój wiedzy o sieciach neuronowych były dwie pozycje: praca [62], prezentująca tę problematykę od strony techniki oraz książka [74], stanowiąca podstawowe źródło wiadomości biologicznych. Już w początkowym okresie zauważono, że cenną własnością sieci neuronowych jest ich zdolność do równoległego przetwarzania informacji, oraz możliwość ich uczenia, zastępująca programowanie.

Pierwszym szeroko znanym przykładem zrealizowanej praktycznie sieci neuronowej był Perceptron [68]. Sieć ta jako układ częściowo elektromechaniczny, a częściowo elektroniczny została zbudowana w 1957 roku przez Franka Rosenblatta i Charlesa

Wightmana. Zadaniem perceptronu było rozpoznawanie znaków alfanumerycznych połączone z procesem uczenia. Ważne było zwłaszcza to, że Perceptron był maszyną zbudowaną fizycznie i działającą w rzeczywistości, co stanowiło wielki krok naprzód w dziedzinie sieci neuronowych w stosunku do wcześniejszych prac, będących bez wyjątku studiami wyłącznie teoretycznymi. Po ogłoszeniu wyników eksperymentalnych badań zachowania tego pierwszego neurokomputera przez jego twórców oraz po potwierdzeniu tych wyników przez ich naśladowców (na przykład na terenie byłego Związku Radzieckiego pierwszy perceptron *Alfa* zbudowany przez Nikołaja Głuszkowa zaczął działać w dwa miesiące po ogłoszeniu pionierskich publikacji Franka Rosenblatta, wkrótce potem w Lyonie zbudowano pierwszą francuską maszynę podobnego typu itd.) nastąpił gwałtowny rozwój tego typu sieci neuronowych na całym świecie.

Ciekawym przykładem sieci neuronowej było rozwiązanie zaproponowane przez Bernarda Widrowa z Uniwersytetu Standforda. Zbudował on w 1960 roku sieć elektrochemicznych uczących się elementów nazywanych Adaline (od *Adaptive linear element*). Był to pierwszy neurokomputer oferowany komercyjnie, wykorzystywany praktycznie do adaptacyjnej filtracji sygnałów echa w długich liniach telekomunikacyjnych.

W latach 70-tych tempo rozwoju badań nad problematyką sztucznych sieci neuronowych zostało gwałtownie zahamowane, a to za sprawą publikacji książki [55], która zniechęciła wielu badaczy, gdyż zawierała formalny dowód na to, że sieci jednowarstwowe (podobne do perceptronu) mają bardzo ograniczony zakres zastosowań. Taki stan zastoju w tej dziedzinie trwał blisko 15 lat i przełamała go dopiero seria prac (Hopfielda, Rumelharta, Sejnowskiego i innych) pokazujących, że nieliniowe sieci **wielowarstwowe** wolne są od ograniczeń wskazanych w pracy [55], a także podających efektywne metody uczenia takich wielowarstwowych sieci neuronowych.

Okres lat 70-tych nie był jednak całkiem bezproduktywny jeśli chodzi o nowe konstrukcje sieci neuronowych. Wymienić tu należy chociażby zbudowaną w 1970 roku przez Stephena Grossberga na uniwersytecie w Bostonie sieć Avalanche służącą do rozpoznawania mowy oraz sterowania ruchami ramienia robota. W tym samym roku w MIT powstaje Cerebellatron skonstruowany przez badaczy Davida Mara, Jamesa Albusa i Andresa Pellioneza, służący także do sterowania robotem. Odmienne zastosowanie miała sieć Brain State in a Box zbudowana przez Jamesa Andersona z uniwersytetu Browna w 1977 roku. Jej zadaniem było wydobywanie wiadomości z bazy danych. Funkcjonalnie sieć ta była odpowiednikiem pamięci asocjacyjnej z dwustronnym dostępem (BAM), ale jej działanie nie

było związane z iteracyjnym procesem poszukiwania, lecz polegało na szybkich zależnościach typu wejście – wyjście.

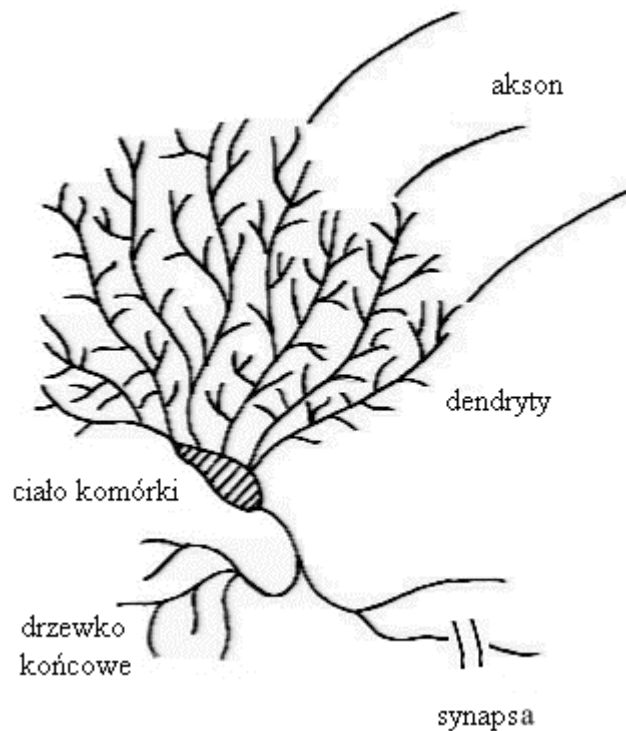
W latach 80-tych powstały pierwsze konstrukcje sieci neuronowych o dużych rozmiarach i znaczących mocach obliczeniowych, a to za sprawą opracowania technologii wytwarzania modeli komórek nerwowych w postaci odpowiednich układów scalonych.

W tym czasie pojawiły się także pierwsze sieci ze sprzężeniami zwrotnymi. Przykładem sieci tego typu są systemy opracowane w 1982 roku przez Johna Hopfielda z AT&T Bell Labs wykorzystywane do odtwarzania obrazów z ich fragmentów, a także do rozwiązywania zadań optymalizacyjnych (słynny problem komiwojżera).

Odrodzenie zainteresowania sieciami neuronowymi nastąpiło w drugiej połowie lat 80-tych po publikacji przełomowej książki J.A. Andersona [1]. Bardzo istotny był także ogromny postęp, jaki dokonał się w dziedzinie technologii układów mogących modelować sieci neuronowe, oraz rozwój technik ich modelowania. Począwszy od końca lat 80-tych datuje się zatem wzrost liczby projektów badawczych w dziedzinie sieci neuronowych. Wzrosła również liczba zagadnień, które można rozwiązywać przy użyciu sieci neuronowych.

2.2. Inspiracje biologiczne

Pierwowzorem sztucznej sieci neuronowej jest biologiczny układ nerwowy. Jest to złożona struktura, której główne elementy to komórki nerwowe (neurony) i połączenia między nimi. Dla pokazania poziomu złożoności należy powiedzieć, że liczba komórek nerwowych w mózgu człowieka jest szacowana na kilkanaście do kilkudziesięciu miliardów komórek, natomiast liczba połączeń między nimi sięga 10^{14} .



Rys. 1. Ogólna budowa komórki nerwowej.

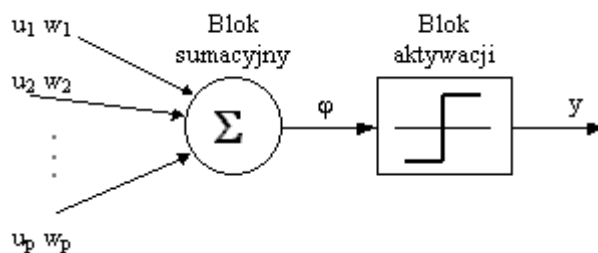
Pojedynczy neuron jest specjalizowaną komórką biologiczną, która tym się odróżnia od innych komórek ciała, że może przenosić i przetwarzać złożone sygnały elektrochemiczne. Neuron składa się z wielu dendrytów stanowiących wejścia informacyjne, z ciała komórki scalającego sygnały ze wszystkich wejść, oraz zawiera wyróżnioną wypustkę, stanowiącą nośnik informacji opuszczających komórkę – akson. Akson danego neuronu połączony jest poprzez synapsy z dendrytami innych neuronów i w ten sposób zbiorowość neuronów tworzy sieć.

Mechanizm działania biologicznej sieci neuronowej jest następujący: każdy z dendrytów dostarcza do komórki sygnał o danym poziomie, zależnym od zbieranych przez dendryt informacji. W komórce następuje skumulowanie tych sygnałów (czasowe i przestrzenne) i jeśli sygnał wyjściowy osiągnie wartość progową, w obszarze tak zwanego wzgórka aksonowego generowany jest krótkotrwały impuls elektryczny, który następnie przewodzony jest przez synapsę do wejścia kolejnej komórki nerwowej. Mamy zatem do czynienia z dwoma stanami komórki nerwowej - pobudzeniem i brakiem aktywności.

2.3. Podstawowy model sztucznego neuronu

Mówiąc o sieciach neuronowych w aspekcie technicznym w istocie mamy na myśli sztuczne sieci neuronowe, będące niezwykle okrojonym modelem rzeczywistości, gdyż jak wynika z badań anatomicznych na ludzki mózg składa się około 10 miliardów komórek nerwowych – neuronów, a sztuczna sieć (modelowana sieć) neuronowa w przeważającej większości posiada nie więcej niż kilkaset takich elementarnych „komórek” – sztucznych neuronów.

Schematycznie sztuczny neuron przedstawia rysunek 2.



Rys. 2. Model sztucznego neuronu.

Sztuczny neuron można zdefiniować w następujący sposób:

- ✓ Do neuronu docierają pewne sygnały czyli wartości wejściowe. Są to albo wartości pierwotne pochodzące z zewnątrz i służące do zadawania danych dla obliczeń wykonywanych przez sieć, albo są to sygnały już częściowo przetworzone i pochodzące z wyjść innych neuronów. Należy przy tym zauważyć, że każdy neuron posiada zależną od potrzeb liczbę n wejść i dokładnie jedno wyjście. Wejścia na rysunku oznaczone są jako u_1 do u_p natomiast wyjście jako y .
- ✓ Wartości docierające do neuronu są przemnażane przez pewien współczynnik liczbowy zwany wagą synaptyczną (w_i). Zazwyczaj dla każdego wejścia waga przyjmuje inną wartość, ustalaną w procesie uczenia. Współczynniki wagowe są zatem odpowiednikiem efektywności transmisji i oddziaływania biologicznego specjalnego hormonu (neuromediatora albo neurotransmitera) w obrębie poszczególnych synaps w neuronie biologicznym.
- ✓ Sposób funkcjonowania neuronu składa się z dwóch faz: wyznaczenie zagregowanej wartości wejściowej, co odbywa się w bloku sumacyjnym oraz wyznaczenie wartości wyjściowej neuronu, co następuje w bloku aktywacji.

- ✓ Wyznaczenie zagregowanej wartości wejściowej polega na obliczeniu ważonej sumy wejść. Niekiedy także (nie we wszystkich rodzajach sieci) do tak utworzonej sumy dodaje się pewien dodatkowy składnik (najczęściej jest to wartość ujemna) zwany wartością progową (r), która określa, jak silne musi być pobudzenie neuronu, by doszło do jego „zapłonu”. Agregacji danych wejściowych dokonuje się więc zwykle zgodnie ze wzorem:

$$\varphi = r + \sum_{i=1}^p u_i w_i \quad (2.1)$$

Uzyskana w ten sposób wartość pomocnicza (φ) określa pobudzenie neuronu.

- ✓ Sygnał reprezentujący łączne pobudzenie neuronu przekształcany jest zazwyczaj przez ustaloną funkcję aktywacji neuronu (zwaną inaczej funkcją przejścia). Wartość obliczona przez funkcję aktywacji jest ostatecznie wartością wyjściową neuronu.

Bardzo duży wpływ na zachowanie tak neuronu jak i całej sieci neuronowej ma zastosowana funkcja aktywacji. W najprostszym przypadku funkcja aktywacji przyjmuje postać funkcji identycznościowej:

$$y = f(\varphi) = \varphi \quad (2.2)$$

Wówczas na wyjściu pojawia się zagregowana wartość wejściowa. Taka funkcja aktywacji definiuje funkcjonowanie sieci liniowych. W literaturze sieci takie nazywane są również sieciami *Madaline* [81], a tworzące je neurony nazywa się odpowiednio *Adaline*. Są to najprostsze ze znanych sieci, które znalazły praktyczne zastosowanie.

Innym typem funkcji opisującej blok aktywacji może być progowa funkcja aktywacji, zwana także funkcją skoku jednostkowego.

$$y = \begin{cases} 1 & \text{dla } \varphi \geq \varphi_h \\ 0 & \text{dla } \varphi < \varphi_h \end{cases} \quad (2.3)$$

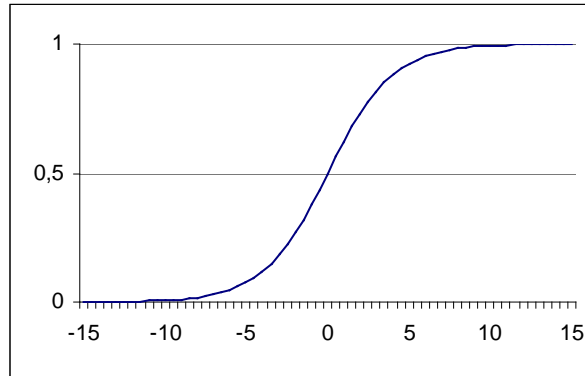
gdzie φ_h jest zadaną stałą wartością progową.

Taki model sztucznego neuronu działa w sposób zbliżony do neuronu biologicznego, co w neurofizjologii określane jest jako zasada „wszystko albo nic”. W rzeczywistości progowa funkcja aktywacji jest rzadko stosowana, gdyż tego rodzaju sieci sprawiają kłopoty podczas uczenia. Z tego też powodu najczęściej stosowane są funkcje aktywacji dostarczające sygnałów o wartościach zmieniających się w sposób ciągły.

Bardzo popularną funkcją przejścia neuronu jest funkcja sigmoidalna opisana wzorem:

$$y = \frac{1}{1 + e^{-\beta\varphi}} \quad (2.4)$$

gdzie β jest zadany parametrem determinującym nachylenie wykresu (dla dużych wartości parametru β wykres funkcji przebiega stromo i upodabnia się do funkcji progowej, natomiast przy niewielkich wartościach parametru β wykres funkcji przypomina wykres funkcji liniowej).



Rys. 3. Wykres funkcji sigmoidalnej.

Funkcja tangensoidalna:

$$y = \operatorname{tgh}\left(\frac{\beta\varphi}{2}\right) = \frac{1 - e^{-\beta\varphi}}{1 + e^{-\beta\varphi}} \quad (2.5)$$

gdzie β ma interpretację taką jak przy funkcji sigmoidalnej. Funkcja tangensoidalna przypomina swoim wyglądem funkcję sigmoidalną, z tą jednak różnicą, że jej wartości znajdują się w przedziale od minus do plus jedności.

Czasami wykorzystuje się także radialne (gaussowskie) funkcje aktywacji, opisywane wzorem:

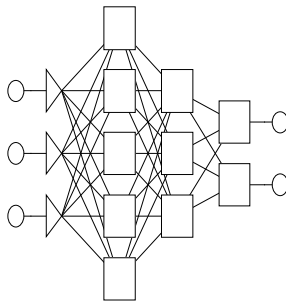
$$y = e^{-\beta\varphi^2} \quad (2.6)$$

Mają one kształt dzwonowy, z maksimum przypadającym zwykle nad wartością zerową². Występujący we wzorze współczynnik β odpowiada za szerokość wykresu funkcji - im większa jest jego wartość, tym wykres funkcji jest bardziej szpiczasty.

² Ewentualne przesunięcie centralnego punktu dzwonowatej funkcji radialnej następuje po dodaniu stałego współczynnika do funkcji zagregowanych sygnałów φ , która ma zresztą w sieciach RBF nieco inną postać.

2.4. Budowa i zasada funkcjonowania sztucznych sieci neuronowych

Możliwości pojedynczego neuronu w zakresie przetwarzania informacji są stosunkowo niewielkie i z tego względu najczęściej stosowane są (podobnie jak w przypadku układów nerwowych organizmów żywych) połączone ze sobą grupy sztucznych neuronów (czyli sieci neuronowe), pozwalające na przeprowadzanie znacznie bardziej złożonych obliczeń.



Rys. 4. Struktura przykładowej sztucznej sieci neuronowej.

Neurony wchodzące w skład sztucznej sieci neuronowej ułożone są najczęściej w warstwach. Wartości wyjściowe wyznaczone dla neuronów jednej warstwy wprowadzane są na wejścia neuronów warstwy następnej. Wyjątkiem jest tu warstwa pierwsza (zwana *warstwą wejściową*) składająca się z neuronów, do których wejść doprowadzane są wartości zmiennych obserwowanych na zewnątrz sieci oraz warstwa ostatnia (*wyjściowa*), składająca się z neuronów wyznaczających wynik obliczeń traktowany jako wartość wyjściową całej sieci. Warstwy znajdujące się pomiędzy warstwą wejściową i wyjściową nazywane są *warstwami ukrytymi*, które wypracowują pewne dane pośrednie, będące podstawą dla procesu wyznaczania ostatecznego rozwiązania.

Występujące w sieci połączenia komunikują zwykle wszystkie neurony jednej warstwy ze wszystkimi neuronami warstwy kolejnej. Taki typ połączeń znany jest pod nazwą połączeń typu *każdy-z-każdym*. Proste sieci neuronowe, które wykorzystywane są w większości przypadków mają strukturę jednokierunkową (*feedforward*), co oznacza, że sygnał przepływa w nich tylko w jednym kierunku: od wejść poprzez neurony ukryte aż do neuronów wyjściowych. Strukturę taką cechuje zawsze stabilne zachowanie, co jest jej dużą zaletą. Jeśli jednak sieć ma wbudowane sprzężenie zwrotne (zawiera połączenia powrotne od

późniejszych do wcześniejszych neuronów), to wówczas może wykonywać bardziej skomplikowane obliczenia, w szczególności o charakterze rekurencyjnym [29]. Wykonano również badania, z których wynika, że sieć o mniejszej liczbie neuronów, posiadająca sprzężenie zwrotne, może dzięki nim wykonywać równie złożone obliczenia, jak sieć bez tych sprzężeń, zawierająca znacznie większą liczbę neuronów. Jednak zastosowanie w sieci sprzężenia zwrotnego ma także pewne słabe strony. Na skutek krążenia sygnałów w sieci ze sprzężeniem zwrotnym (z wejścia do wyjścia i przez sprzężenie zwrotne z powrotem na wejście) może ona zachowywać się niestabilnie i mieć bardzo złożoną dynamikę, np. w formie chaosu deterministycznego. W związku z tym sieci rekurencyjne są obiektem zainteresowania naukowców, a wśród praktyków wykorzystywane są raczej sieci o strukturze jednokierunkowej.

Sieć neuronowa w trakcie swojego działania przetwarza wprowadzone na jej wejścia wartości zmiennych wejściowych w wyniku czego uzyskiwane są na jej wyjściach wartości zmiennych wyjściowych. Sposób pracy sieci uzależniony jest od wielu czynników, do których należy zaliczyć przede wszystkim:

1. Przyjęte modele neuronów - a więc wybrane dla neuronów sieci konkretne postacie funkcji agregującej i funkcji aktywacji; zwykle wszystkie neurony wchodzące w skład tej samej warstwy korzystają z takich samych formuł funkcji agregującej i funkcji aktywacji, natomiast w różnych warstwach stosowane są często neurony korzystające z różniących się formuł;
2. Wartości współczynników wagowych neuronów - są one ustalane automatycznie w trakcie procesu uczenia i dlatego są różne dla poszczególnych wejść i dla poszczególnych neuronów;
3. Liczba warstw sieci;
4. Liczba neuronów w poszczególnych warstwach sieci;
5. Przyjęty sposób połączeń neuronów.

Warunkiem koniecznym do właściwego funkcjonowania sieci jest poprawne określenie wszystkich wymienionych powyżej czynników.

2.5. Metody uczenia sztucznych sieci neuronowych

Pewne podobieństwa pomiędzy rzeczywistymi i sztucznymi sieciami neuronowymi można dostrzec nie tylko w ich budowie, ale również w sposobie pozyskiwania wiedzy niezbędnej do ich prawidłowego funkcjonowania. Proces pozyskiwania wiedzy przez sieć neuronową musi gwarantować możliwość późniejszego jej uogólniania w celu rozwiązania całego postawionego przed nią zadania (a nie tylko odtworzenia wiedzy zgromadzonej w tzw. zbiorze uczącym).

Proces przygotowania sieci do prawidłowego działania nazywany jest uczeniem. Sieć uczy się prawidłowo działać na podstawie prezentowanych jej przykładów. Bazując na przedstawionych rzeczywistych przypadkach sieć stara się odkryć i zapamiętać ogólne prawidłowości charakteryzujące te obiekty. Rozpoznane reguły sztuczna sieć neuronowa przechowuje w postaci rozproszonej w wartościach współczynników wagowych neuronów. Proces uczenia polega zatem na prawidłowym określeniu wartości współczynników wagowych neuronów na podstawie informacji wydobytych w trakcie procesu uczenia ze zbioru uczącego.

Kluczowym elementem w procesie uczenia są **wagi** wejść poszczególnych neuronów. Jeśli zmieniają się wartości wag - neuron zacznie pełnić innego rodzaju funkcję w sieci, a co za tym idzie cała sieć zacznie inaczej działać. Uczenie sieci polega więc na tym, by tak dobrać wagi, aby wszystkie neurony wykonywały dokładnie takie czynności, jakich się od nich wymaga.

Uczenie sieci rozpoczyna się od nadania wagom neuronów wartości losowych. W każdym kroku iteracyjnego procesu uczenia wartości wag jednego lub kilku neuronów ulegają zmianie, przy czym reguły tych zmian są tak pomyślane, by każdy neuron sam potrafił określić, które ze swoich wag ma zmienić, w którą stronę, a także o ile. Możliwe są dwa warianty procesu uczenia:

- *z nauczycielem*
- *bez nauczyciela.*

Uczenie z nauczycielem polega na tym, że sieci podaje się przykłady poprawnego działania, które powinna ona potem naśladować w swoim bieżącym działaniu (w czasie egzaminu). Przykład należy rozumieć w ten sposób, że nauczyciel podaje konkretne sygnały wejściowe i wyjściowe, pokazując jaka jest wymagana odpowiedź sieci dla pewnej konfiguracji danych wejściowych. Mamy do czynienia z parą wartości - przykładowym sygnałem wejściowym i pożądanym (oczekiwanym) wyjściem, czyli wymaganą odpowiedzią

sieci na ten sygnał wejściowy. Zbiór przykładów zgromadzonych w celu ich wykorzystania w procesie uczenia sieci nazywa się zwykle ciągiem uczącym. Zatem w typowym procesie uczenia sieć otrzymuje od nauczyciela ciąg uczący i na jego podstawie uczy się prawidłowego działania, stosując jedną z wielu znanych dziś strategii uczenia.

Obok opisanego wyżej schematu uczenia z nauczycielem występuje też szereg metod tak zwanego uczenia bez nauczyciela (albo samouczenia sieci). Metody te polegają na podawaniu na wejście sieci wyłącznie szeregu przykładowych danych wejściowych, bez podawania jakiegokolwiek informacji dotyczącej pożądanych czy chociażby tylko oczekiwanych sygnałów wyjściowych. Odpowiednio zaprojektowana sieć neuronowa potrafi wykorzystać same tylko obserwacje wejściowych sygnałów i zbudować na ich podstawie sensowny algorytm swojego działania - najczęściej polegający na tym, że automatycznie wykrywane są klasy powtarzających się sygnałów wejściowych i sieć uczy się (spontanicznie, bez jawnego nauczania) rozpoznawać te typowe wzorce sygnałów.

Samouczenie jest też bardzo interesujące z punktu widzenia zastosowań, gdyż nie wymaga żadnej jawnie podawanej do sieci neuronowej zewnętrznej wiedzy, a sieć zgromadzi wszystkie potrzebne informacje i wiadomości.

Poniżej zostaną omówione dwie podstawowe reguły uczenia: reguła Delta leżąca u podstaw większości algorytmów uczenia z nauczycielem oraz reguła Hebba stanowiąca przykład uczenia bez nauczyciela.

2.5.1. Reguła Delta

Reguła delta jest regułą uczenia z nauczycielem, wprowadzoną przez Widrowa i Hoffa [81]. Polega ona na tym, że każdy neuron po otrzymaniu na swoich wejściach określonych sygnałów wyznacza swój sygnał wyjściowy, wykorzystując posiadaną wiedzę w postaci wcześniej ustalonych wartości współczynników wagowych wszystkich wejść oraz progu. Wartość sygnału wyjściowego, wyznaczonego przez neuron na danym kroku procesu uczenia porównywana jest z odpowiedzią wzorcową podaną przez nauczyciela w ciągu uczącym. Jeśli występuje rozbieżność - neuron wyznacza różnicę pomiędzy swoim sygnałem wyjściowym a tą wartością sygnału, która była by - według nauczyciela prawidłowa. Ta różnica oznaczana jest zwykle symbolem greckiej litery δ (delta) i stąd nazwa opisywanej metody.

Sygnal błędu (delta) wykorzystywany jest przez neuron do korygowania swoich współczynników wagowych, stosując następujące reguły:

- wagi zmieniane są tym silniej, im większy błąd został wykryty;
- wagi związane z tymi wejściami, na których występowały duże wartości sygnałów wejściowych, zmieniane są bardziej, niż wagi wejść, na których sygnał wejściowy był niewielki.

Znając błąd popełniony przez neuron oraz jego sygnały wejściowe możemy łatwo przewidzieć, jak będą się zmieniać jego wagi. Rozważaną tu regułę delta można zapisać w postaci następującego wzoru:

$$W^{(j+1)} = W^{(j)} + \mu^{(j)} \delta^{(j)} X^{(j)} \quad (2.7)$$

We wzorze tym:

$$\delta^{(j)} = z^{(j)} - y^{(j)}$$

$$y^{(j)} = W^{(j)} X^{(j)}$$

gdzie: μ to współczynnik liczbowy decydujący o szybkości uczenia, $\delta^{(j)}$ to wartość błędu w j-tym kroku uczenia, z to wymagana, a y to rzeczywista odpowiedź neuronu na sygnał x , $W^{(j)}$ to wektor wag w j-tym kroku uczenia.

Sieć stosując opisane metody w praktyce sama przerywa proces uczenia gdy jest już dobrze wytrenowana, gdyż małe błędy powodują jedynie minimalne korekty wag. Jest to logiczne, podobnie jak zasada uzależniania wielkości korekty od wielkości wejściowego sygnału przekazywanego przez rozważaną wagę, ponieważ jeśli odpowiednia składowa była mała to korygowana wartość wagi w niewielkim tylko stopniu przyczyniła się do powstania usuwanego błędu. W szczególności opisany algorytm powoduje, że dla wejść, na których w danym momencie nie były podawane sygnały (wejścia o zerowych wartościach wag) odpowiednie wagi nie są zmieniane, nie wiadomo bowiem, czy są dobre czy nie, gdyż nie uczestniczyły w tworzeniu aktualnego sygnału wyjściowego.

W praktycznych realizacjach opisanego wyżej algorytmu dochodzi kilka dalszych godnych uwagi elementów. W pierwszej kolejności twórca sieci musi zdecydować, jak silne powinny być zmiany wag powodowane przez określone wartości sygnałów wejściowych i określoną wielkość błędu. Ten współczynnik proporcjonalności η , zwany *learning rate*, może być wybierany dowolnie, jednak każda konkretna decyzja dotycząca jego wartości ma określone konsekwencje. Wybranie współczynnika za małego prowadzi do bardzo powolnego procesu uczenia (wagi są poprawiane w każdym kroku bardzo słabo, żeby więc osiągnęły

pożądane wartości trzeba wykonać bardzo dużo takich kroków). Z kolei wybór za dużego współczynnika uczenia powoduje bardzo gwałtowne zmiany parametrów sieci, które w krańcowym przypadku prowadzić mogą nawet do niestabilności procesu uczenia. Konieczny jest więc kompromisowy wybór współczynnika uczenia, uwzględniający zarówno korzyści związane z szybką pracą, jak i względy bezpieczeństwa, wskazujące na konieczność uzyskania stabilnej pracy procesu uczenia.

2.5.2. Reguła Hebba

Jest to jedna z najpopularniejszych metod samouczenia sieci neuronowych. Polega ona na tym, że sieci pokazuje się kolejne przykłady sygnałów wejściowych, nie podając żadnych informacji o tym, co z tymi sygnałami należy zrobić. Sieć obserwuje otoczenie i odbiera różne sygnały. Nikt nie określa jednak, jakie znaczenie mają pokazujące się obiekty i jakie są pomiędzy nimi zależności. Sieć na podstawie obserwacji występujących sygnałów stopniowo sama odkrywa, jakie jest ich znaczenie i również sama ustala zachodzące między sygnałami zależności.

Po podaniu do sieci neuronowej każdego kolejnego zestawu sygnałów wejściowych tworzy się w niej pewien rozkład sygnałów wyjściowych - niektóre neurony sieci są pobudzone bardzo silnie, inne słabiej, a jeszcze inne mają sygnały wyjściowe wręcz ujemne. Interpretacja tych zachowań może być taka, że niektóre neurony „rozpoznają” podawane sygnały jako „własne” (czyli takie, które są skłonne akceptować), inne traktują je „obojętnie”, zaś jeszcze u innych neuronów wzbudzają one wręcz „awersję”.

Po ustaleniu sygnałów wyjściowych wszystkich neuronów w całej sieci - wszystkie wagi wszystkich neuronów są zmieniane, przy czym wielkość odpowiedniej zmiany wyznaczana jest na podstawie iloczynu sygnału wejściowego, wchodzącego na dane wejście neuronu i sygnału wyjściowego produkowanego przez neuron, w którym modyfikujemy wagi. Można to formalnie zapisać w postaci wzoru:

$$w_i^{(m)(j+1)} = w_i^{(m)(j)} + \mu x_i^{(j)} y_m^{(j)} \quad (2.8)$$

przy czym:

$$y_m^{(j)} = \sum_{i=1}^n w_i^{(m)(j)} x_i^{(j)}$$

gdzie: μ to współczynnik liczbowy decydujący o szybkości uczenia, y to rzeczywista odpowiedź neuronu na sygnał x . Podwójne górne indeksy przy współczynnikach wagowych $w_i^{(m)(j)}$ wynikają z faktu, że trzeba uwzględnić numerację neuronów, do których wagi te należą (m) oraz numerację kroków wynikającą z kolejnych pokazów.

Łatwo zauważyć, że jest to właśnie realizacja postulatu Hebb'a [30]. W efekcie opisanego wyżej algorytmu połączenia między źródłami silnych sygnałów i neuronami które na nie silnie reagują są wzmacniane.

Dokładniejsza analiza procesu samouczenia metodą Hebb'a pozwala stwierdzić, że w wyniku konsekwentnego stosowania opisanego algorytmu początkowe, najczęściej przypadkowe „preferencje” neuronów ulegają systematycznemu wzmacnianiu i dokładnej polaryzacji. Jeśli jakiś neuron miał „wrodzoną skłonność” do akceptowania sygnałów pewnego rodzaju - to w miarę kolejnych pokazów nauczy się te sygnały rozpoznawać coraz dokładniej i coraz bardziej precyzyjnie. Po dłuższym czasie takiego samouczenia w sieci powstaną zatem wzorce poszczególnych typów sygnałów występujących na wejściu sieci. W wyniku tego procesu sygnały podobne do siebie będą w miarę postępu uczenia coraz skuteczniej grupowane i rozpoznawane przez pewne neurony, zaś inne typy sygnałów staną się „obiektem zainteresowania” innych neuronów. W wyniku tego procesu samouczenia sieć nauczy się, ile klas podobnych do siebie sygnałów pojawia się na jej wejściach oraz sama przyporządkuje tym klasom sygnałów neurony, które nauczą się je rozróżniać, rozpoznawać i sygnalizować.

Proces samouczenia ma niestety wady. W porównaniu z procesem uczenia z nauczycielem samouczenie jest zwykle znacznie powolniejsze. Co więcej bez nauczyciela nie można z góry określić, który neuron wyspecjalizuje się w rozpoznawaniu której klasy sygnałów. Stanowi to pewną trudność przy wykorzystywaniu i interpretacji wyników pracy sieci. Co więcej - nie można określić, czy sieć uczona w ten sposób nauczy się wszystkich prezentowanych jej wzorców. Dlatego sieć przeznaczona do samouczenia musi być większa niż sieć wykonująca to samo zadanie, ale trenowana w sposób klasyczny, z udziałem nauczyciela. Szacunkowo sieć powinna mieć co najmniej trzykrotnie więcej elementów warstwy wyjściowej niż wynosi oczekiwana liczba różnych wzorców, które sieć ma rozpoznawać.

Bardzo istotną kwestią w przypadku samouczenia jest wybór początkowych wartości wag neuronów. Wartości te mają bardzo silny wpływ na ostateczne zachowanie sieci. Wynika to z faktu, iż proces uczenia jedynie pogłębia i doskonali pewne tendencje istniejące w sieci od samego początku, przeto od jakości tych początkowych, „wrodzonych” właściwości sieci silnie zależy, do czego sieć dojdzie na końcu procesu uczenia. Nie wiedząc z góry, jakiego zadania sieć powinna się uczyć, trudno wprowadzać jakikolwiek zdeterminowany mechanizm nadawania początkowych wartości wag, jednak pozostawienie wszystkiego wyłącznie mechanizmom losowym może powodować, że sieć (zwłaszcza mała) może nie zdołać

wystarczająco zróżnicować swego działania w początkowym okresie procesu uczenia i wszelkie późniejsze wysiłki, by znaleźć w strukturze sieci reprezentację dla wszystkich występujących w wejściowych sygnałach klas, mogą okazać się daremne.

Można jednak wprowadzić pewien mechanizm wstępnego doboru wartości wag w początkowej fazie procesu uczenia. Metoda ta, zwana *convex combination* modyfikuje początkowe wartości wag w taki sposób, by zwiększyć prawdopodobieństwo równomiernego pokrycia przez poszczególne neurony wszystkich typowych sytuacji pojawiających się w wejściowym zbiorze danych. Jeśli tylko dane pojawiające się w początkowej fazie uczenia nie będą różniły się istotnie od tych, jakie sieć będzie potem analizować i różnicować - metoda *convex combination* stworzy w sposób automatyczny dogodny punkt wyjścia do dalszego samouczenia i zapewni stosunkowo dobrą jakość nauczonej sieci w większości praktycznych zadań.

Proces uczenia sieci uwalnia nas od uciążliwego tworzenia i zapisywania (w określonym języku programowania) algorytmu wymaganego dla takiego przetwarzania wejściowych danych, by uzyskać pożądaną końcowy. Nie odbywa się to jednak za darmo, gdyż ceną, jaką trzeba zapłacić za tę wygodę jest długotrwały i wymagający sporych mocy obliczeniowych proces uczenia. Zwykle jednokrotna prezentacja wszystkich przypadków wchodzących w skład zbioru uczącego (czyli tzw. jedna epoka uczenia) nie wystarcza do osiągnięcia prawidłowego działania sieci. Dlatego też dane uczące prezentowane są wielokrotnie – często kilkaset, kilka tysięcy, albo nawet milionów razy. Co gorsza, proces uczenia jest zawsze procesem indeterministycznym, co oznacza, że wynik uczenia nie jest nigdy całkowicie pewny. Czasami ta sama sieć w tym samym zadaniu może uzyskiwać znacząco różne (co do jakości) rozwiązania postawionego problemu. Fakt ten powoduje, że dla osiągnięcia najlepszego możliwego wyniku proces uczenia trzeba niekiedy powtarzać kilkakrotnie, za każdym razem startując od innych wartości początkowych przyjętych dla współczynników wag. Jest to bardzo kłopotliwe, ale czasami bywa jedynym sposobem pokonania pojawiających się trudności. Na szczęście trudności te nie zawsze występują, co powoduje, że wśród użytkowników sieci neuronowe cieszą się opinią narzędzia sprawnego i wygodnego w użyciu.

3. Modele prognostyczne

3.1. Dobór zmiennych wejściowych

Dobór zmiennych wejściowych w każdym modelu decyzyjnym, prognostycznym czy ekonometrycznym opartym zarówno na sieciach neuronowych jak i innych narzędziach - jest sprawą kluczową. Od odpowiedniego doboru wektora wejściowego w dużej mierze zależą możliwości prognostyczne zbudowanego modelu. Nie ulega wątpliwości, że istnieje wiele czynników kształtujących cenę waloru czy cenę koszyka akcji (wartość indeksu), zarówno z grupy tzw. wskaźników technicznych (wartości otrzymane w wyniku analizy zmienności kursów, wolumenu obrotów), danych opóźnionych (ceny akcji z poprzednich okresów), wartości indeksów giełd zagranicznych, wskaźników makroekonomicznych (kursy walut, wysokość stóp procentowych, itp.), oraz wielu innych, często nie dających się bezpośrednio zmierzyć (na przykład poziom optymizmu inwestorów). Wybór jednych, a odrzucenie innych zmiennych jest niezwykle trudne, aczkolwiek kryteria są dość oczywiste. Wybieramy te zmienne, które są silnie skorelowane ze zmienną prognozowaną, ale jednocześnie możliwie słabo skorelowane między sobą.

Program *Statistica Neural Networks*, używany w tej pracy jako podstawowe narzędzie badawcze, może być bardzo pomocny przy doborze optymalnego wektora wejściowego, ponieważ istnieje w nim wiele dostępnych metod minimalizowania zbioru zmiennych niezależnych, jak chociażby algorytm genetyczny czy analiza wrażliwości. Warunek słabego skorelowania zmiennych objaśniających między sobą jest szczególnie istotny przy budowie modeli ekonometrycznych, estymowanych za pomocą klasycznej metody najmniejszych kwadratów, gdyż bez jego zachowania otrzymuje się bardzo duże wariancje estymatorów parametrów, co znacznie pogarsza jakość oszacowanych modeli [85]. Natomiast w przypadku modeli neuronowych skorelowanie zmiennych między sobą ma znacznie mniejszy negatywny wpływ na jakość modeli dzięki naturalnej możliwości sieci neuronowych do osłabiania wpływu niektórych wejść, poprzez minimalizację ich współczynników wagowych. Jednak także i w tym przypadku usunięcie ze zbioru danych zbędnych zmiennych objaśniających daje inną niezmiernie ważną korzyść, a mianowicie minimalizację wymiaru przestrzeni wejść. Jest to o tyle istotne, że budując modele decyzyjne czy prognostyczne zwłaszcza w zastosowaniach ekonomicznych, zawsze mamy kłopoty z liczebnością zbioru obserwacji służących do estymacji parametrów modelu. Często niektóre dane są po prostu niedostępne lub ich uzyskanie może być nadmiernie kosztowne. Co prawda ten problem nie występuje (pozornie) w przypadku notowań giełdowych, jednak zwiększanie ich liczby może

powodować, że część z nich, szczególnie tych odległych w czasie, uległa dezaktualizacji. Gdy budujemy model neuronalny to zależy nam na tym, aby uzyskać jak najmniejszy błąd sieci związany z prezentacją nowego przypadku (takiego, który nie był prezentowany w trakcie uczenia). Zatem ucząc sieć neuronową dążymy do tego, by w przyszłości potrafiła ona uogólnić nabytą wiedzę na nieznaną jej, choć podobne problemy. Tę zdolność nazywa się generalizacją.

Na jakość generalizacji mają wpływ trzy zasadnicze czynniki:

- długość i jakość ciągu uczącego,
- architektura (budowa) sieci,
- złożoność problemu.

Na ostatni z tych czynników nie mamy żadnego wpływu. Pozostaje nam zatem możliwość manipulowania tylko dwoma pierwszymi czynnikami i to w taki sposób, aby dopasować liczbę przypadków do architektury, bądź architekturę (liczbę elementów ukrytych) do długości ciągu uczącego.

Miara zdolności uogólniania sieci jest pojęciem trudnym do zdefiniowania i jest oparta na zależnościach statystycznych, odnoszących się do zbiorów [77]. Podstawową wielkością jest tu miara Vapnika-Chervonenkisa, zwana w skrócie VCdim. Miara VCdim systemu została zdefiniowana jako liczebność największego zbioru S danych wzorców, dla których system może zrealizować wszystkie możliwe 2^n dychotomii liniowych zbioru S (podział zbioru na 2 części za pomocą formuł liniowych, czyli w k -wymiarowej przestrzeni – za pomocą hiperpłaszczyzn $(k-1)$ -wymiarowych).

Dla przykładu VCdim dla neuronu o dwóch wejściach wynosi $n=3$. Łatwo można wykazać, że zbiór złożony z trzech danych uczących jest największym zbiorem, w którym można przeprowadzić podział na dwie liniowo separowalne grupy na 2^3 sposobów. Uogólniając ten wynik stwierdzamy, że dla neuronu o N wejściach miara VCdim wynosi $N+1$. Zatem miara VCdim oznacza maksymalną liczbę danych uczących, które mogą zostać bezbłędnie odtworzone we wszystkich możliwych konfiguracjach. Jednak o ile łatwo jest wyznaczyć wartość VCdim dla pojedynczego neuronu, o tyle w przypadku sieci neuronowej nie jest to już takie proste. Dodatkowo dla każdego rodzaju sieci jest to oddzielny problem, a szczególnie jaszkrawo występuje on w przypadku sieci wielowarstwowych, gdzie liczba wag jest zazwyczaj bardzo duża w stosunku do liczby neuronów.

Główną trudność określenia optymalnego wymiaru zbioru uczącego dla danej sieci wynika zatem z niemożności określenia dokładnej wartości miary VCdim. W ogólnym przypadku jej wartość wzrasta przy wzrastającej liczbie wag. Zbliżanie wartości miary VCdim do liczby danych zawartych w zbiorze uczącym gwałtownie zwiększa prawdopodobieństwo pojawienia się przy uczeniu efektu tzw. uczenia na pamięć. Dlatego też w praktyce dla uzyskania dobrych zdolności generalizacji sieci należy ograniczać liczbę neuronów ukrytych, oraz liczbę powiązań międzyneuronowych, a także stosować takie metody wstępnego przetwarzania danych, które umożliwiają zmniejszenie wymiarowości wektora wejściowego sieci.

Każdy z tych czynników pośrednio lub bezpośrednio wpływa na zmniejszenie efektywnej liczby wag sieci neuronowej. W przypadku występowania nadmiarowych współczynników wagowych, parametry sieci niepotrzebne dla odzwierciedlenia cech modelowanego procesu dopasowują się do wszelkich nieregularności danych uczących, traktując je jako cechę główną. W związku z tym w procesie testowania sieci na danych testujących powstają dodatkowe błędy.

Podstawowy problem stanowi zatem oszacowanie miary Vapnika-Chervonenkisa dla sieci o określonej strukturze. Ogólnie stanowi ona funkcję złożoności sieci, zwłaszcza wag synaptycznych łączących neurony między sobą. Im większa jest liczba różnych wag, tym większa złożoność sieci i wartość miary VCdim. Niestety nie istnieje prosty związek między architekturą sieci wielowarstwowej, a miarą VCdim. Można podać jedynie oszacowanie górnego i dolnego zakresu tej miary [77][39] w postaci:

$$2 \left\lceil \frac{K}{2} \right\rceil N \leq VC \dim \leq 2N_w (1 + \log N_n) \quad (3.1)$$

gdzie: $\lceil \cdot \rceil$ oznacza część całkowitą liczby, N – wymiar wektora wejściowego, K – liczba neuronów w warstwie ukrytej, N_w – całkowita liczba wag w sieci, N_n – całkowita liczba neuronów w sieci.

Jak widać z powyższej zależności, dolna granica przedziału jest w przybliżeniu równa liczbie wag łączących warstwę wejściową z warstwą ukrytą, górna granica natomiast jest większa niż dwukrotna liczba wszystkich wag sieci. W związku z niemożliwością dokładnego wyznaczenia miary VCdim przyjmuje się jej zgrubne oszacowanie w postaci liczby wszystkich wag występujących w sieci neuronowej. Zagadnienie staje się jednak bardziej skomplikowane, jeśli zostanie zastosowana sigmoidalna funkcja aktywacji. W takim przypadku według [39] za oszacowanie miary VCdim przyjęć należy podwójną liczbę wszystkich wag występujących w sieci.

Oszacowanie miary VCdim umożliwia ocenę minimalnego wymiaru zbioru uczącego. W wyniku przeprowadzenia wielu testów numerycznych stwierdzono dobre zdolności uogólniania sieci, jeśli liczba próbek uczących jest minimum 10 razy większa od miary VCdim [39].

Przy rozwiązywaniu większości bardziej złożonych problemów technicznych nie uzyskuje się dobrych rezultatów uogólniania przez manipulowanie tylko danymi uczącymi. Konieczne jest również aktywne dopasowanie stopnia złożoności sieci czyli jej struktury do rodzaju rozwiązywanego problemu. W większości przypadków do rozwiązywania problemów przy pomocy sieci neuronowych używa się sieci najprostszej, czyli zawierającej jedną warstwę ukrytą. Dobór liczby neuronów w tej warstwie jest zagadnieniem kluczowym i decydującym o właściwościach uogólniających sieci.

Są tu możliwe dwa kierunki działań. W pierwszym zakłada się wstępną liczbę neuronów ukrytych, opartą bądź na teorii Kołmogorowa, bądź na dotychczasowych doświadczeniach i wówczas jako dobre przybliżenie można przyjąć liczbę neuronów w warstwie ukrytej jako średnią geometryczną wymiarów wejść i wyjść sieci:

$$K \approx \sqrt{NM} \quad (3.2)$$

Odwrotne podejście polega na starcie z minimalną (0 lub 1) liczbą neuronów ukrytych i stopniowym ich dodawaniu aż do uzyskania dobrego stopnia wytrenowania sieci na zbiorze uczącym.

3.2. Charakterystyka wektora wejściowego

Do dalszych rozważań jako wektor zmiennych wejściowych zostało wybranych 56 zmiennych niezależnych w skład których wchodzi zarówno opóźnione wartości zmiennej wyjściowej, indeksy różnych giełd, a także wybrane wskaźniki analizy technicznej i ceny walut obcych.

Te dane to:

- WIG20 O, WIG20 O t-1, WIG20 O t-2, WIG20 O t-3 – wartość indeksu WIG20 na otwarciu w chwili t, t-1, oraz t-2 i t-3,
- WIG20 H – maksymalna wartość indeksu na danej sesji,
- WIG20 L – minimalna wartość indeksu na danej sesji,
- WIG20 C, WIG20 C t-1, WIG20 C t-2, WIG20 C t-3, - wartość indeksu WIG20 na zamknięciu sesji w czasie t, t-1, t-2, t-3.

- obrót w mln zł oraz średnia obrotu z 5 okresów w mln zł,
- ROC 5, ROC 10 – Rate Of Change – wskaźnik zmian - Jest to wskaźnik impetu mierzący wielkość zmiany ceny w zadanym okresie. Kierunek notowań powinien być potwierdzony analogiczną tendencją wskaźnika. Jeżeli wskaźnik osiąga nowe minimum i zaczyna rosnąć jest to wstępny sygnał kupna, osiągnięcie nowego szczytu i następujący po nim spadek jest wstępnym sygnałem sprzedaży.

$$ROC = \frac{P_n}{P_{n-k}} - 1 \quad (3.3)$$

gdzie: P_{n-k} to kurs sprzed k notowań.

- RSI 9, RSI 14 – Relative Strength Index – indeks siły względnej. Wskaźnik ten został zaprezentowany przez Johna Wellesa Wildera [79] - Jest to jeden z najpopularniejszych i najczęściej używanych oscylatorów. Nie jest to jednak, jak wskazuje nazwa, typowy miernik siły względnej. Jak klasyczny oscylator zyskuje wartości w przedziale 0 -100. Podstawowa analiza zakłada poszukiwanie dywergencji względem wykresu cenowego. RSI jest również miernikiem stanów wykupienia/wyprzedania rynku. Wartości powyżej 70 pkt., są odbierane jako wykupienie rynku, poniżej 30 pkt. jako wyprzedanie rynku. Dodatkowo wskaźnik może być poddawany klasycznym technikom analitycznym, jak kreślenie linii trendów, poszukiwanie formacji cenowych oraz sygnałów płynących z ich przebicia.

$$RSI = 1 - \frac{1}{1 + \frac{U_k}{D_k}} \quad (3.4)$$

gdzie: U_k to średni wzrost kursu z k sesji, a D_k to średni spadek kursu z k sesji.

- CCI 9, CCI 14 - Commodity Channel Index. Wskaźnik CCI został przedstawiony w piśmie „Futures” w 1980 roku przez Donalda Lamberta [46]. Pierwotnie wskaźnik ten był stosowany do identyfikacji cykli na rynku towarów. Wskaźnik CCI z reguły porusza się w przedziale od -100 do 100. Powyżej i poniżej tych granic znajdują się odpowiednio strefa wykupienia i wyprzedania. Lambert zakładał, że ruchy w tym przedziale są przypadkowe i nie mogą stanowić podstawy do podejmowania decyzji inwestycyjnych. Lambert uważał, że jeśli CCI przełamie poziom 100 (-100) to należy otworzyć długą pozycję (krótką pozycję) i zamknąć ją gdy CCI powróci pod (nad) poziom 100 (-100).

$$CCI = \frac{\left(\frac{1}{3}(P_C + P_H + P_L) - SK\left(\frac{1}{3}(P_C + P_H + P_L); n\right) \right)}{\frac{1}{n} \sum_{i=0}^{n-1} \left| \frac{1}{3}(P_C + P_H + P_L)_i - SK\left(\frac{1}{3}(P_C + P_H + P_L); n\right) \right|} * 0,015 \quad (3.5)$$

gdzie: P_C , P_H , P_L to odpowiednio: cena zamknięcia, cena najwyższa i cena najniższa sesji, a $SK(a,n)$ to średnia krocząca dla argumentu z n okresów.

- ATR 14 - Average True Range - średnia rzeczywistego zasięgu. Jest to wskaźnik mierzący zmienność cen. Niskie wartości wskaźnika obecne są w okresach konsolidacji jaka towarzyszy kształtowaniu szczytów np. niektórych formacji cenowych lub bezpośrednio po okresach stabilizacji. Wysokie wartości wskaźnika często towarzyszą kształtowaniu lokalnych minimów poprzedzonych gwałtownymi spadkami.

$$ATR = SK\left(\max(P_H - P_L; P_{C,t-1} - P_H; P_{C,t-1} - P_L), n\right) \quad (3.6)$$

gdzie: $P_{C,t-1}$, P_H , P_L to odpowiednio: cena zamknięcia na sesji w czasie t-1, cena najwyższa i cena najniższa sesji, a $SK(a,n)$ to średnia krocząca dla argumentu z n okresów.

- EMA 5, EMA 9, EMA 12, EMA 26 - Exponential Moving Average. Średnia EMA stosowana jest po to aby częściowo wyeliminować opóźnienia, które wynikają ze stosowania zwykłej średniej kroczącej. Opóźnienia te zredukowane są poprzez nadanie większego znaczenia ostatnim ruchom na rynku. To znaczenie zależy od długości liczonej średniej EMA. Im krótszy okres tym większe znaczenie mają ostatnie zmiany.

$$EMA_t = P_t \left(\frac{2}{n+1} \right) + EMA_{t-1} \left(1 - \frac{2}{n+1} \right) \quad (3.7)$$

gdzie: P_t to dzisiejsza cena waloru, n to okres dla jakiego liczona jest średnia.

- MA 9, MA 12 – średnia z wartości indeksu WIG20 na zamknięciu z odpowiednio 9 i 12 kolejnych sesji.
- WIG O, WIG H, WIG L, WIG C – wartości indeksu WIG.
- SP500 O, SP500C – wartości amerykańskiego indeksu SP500.
- NASDAQ O, NASDAQ C, NASDAQ C t-1, NASDAQ C t-2 – wartości amerykańskiego indeksu NASDAQ.
- DAX O, DAX C, DAX C t-1 – wartości niemieckiego indeksu DAX.

- BUX O, BUX C t-1, BUX C – wartości węgierskiego indeksu BUX.
- Cenę dolara oraz franka szwajcarskiego.
- Cztery ostatnie logarytmiczne jednodniowe stopy zwrotu oraz dwie kolejne ostatnie logarytmiczne dwudniowe stopy zwrotu.
- Nachylenie pięciodniowego trendu³ kursu: obecne i sprzed czterech dni, obecne nachylenie dziesięciodniowego trendu.
- Dynamika zmiany logarytmicznych jednodniowych stóp zwrotu (różnicę tych stóp), dynamika zmiany logarytmicznych dwudniowych stóp zwrotu, dynamika zmiany pięciodniowego trendu (różnica wskaźników trendu obecnego i sprzed 4 dni), oraz różnica między nachyleniem trendu 5-dniowego i 10-dniowego.

Zbiór danych wejściowych składa się z 2230 obserwacji i obejmuje notowania z okresu: 01.01.1997 – 28.11.2005. Zbiór obserwacji został podzielony wg schematu: 1338 – 446 – 446 na podzbiory: uczący, walidacyjny i testowy.

3.3. Optymalizacja wektora zmiennych wejściowych

Niniejszy rozdział zostanie poświęcony metodom optymalnego doboru wektora zmiennych wejściowych do modeli neuronowych przy pomocy pakietu Statistica. Program Statistica Neural Networks daje użytkownikowi wiele różnych algorytmów optymalizacji wektora wejściowego takich jak:

- Analiza głównych składowych
- Analiza wrażliwości
- Automatyczny projektant sieci
- Algorytm genetyczny
- Nieliniowa kompresja danych
- Selekcja krokowa wsteczna i postępująca

Optymalizacja wektora zmiennych wejściowych ułatwia użytkownikowi podjęcie decyzji, które ze zmiennych niezależnych są najbardziej istotne i mają decydujący wpływ na jakość modelu prognostycznego. Jeśli bowiem przyjmemy założenie, że model rynku papierów wartościowych jest nieliniowy to minimalizując wymiar tej przestrzeni możemy

³ Nachylenie trendu było obliczone jako współczynnik kierunkowy prostej regresji dla pięciu ostatnich logarytmów notowań.

spodziewać się minimalizacji błędu predykcji. Dzieje się tak również dlatego, że doboru zmiennych niezależnych dokonuje badacz w sposób subiektywny, a zaszumienie niektórych danych może być na tyle istotne, że zamiast zmniejszania błędu predykcji, będziemy mieli do czynienia z jego wzrostem.

3.3.1. Wyznaczenie optymalnej sieci dla 56 zmiennych wejściowych.

W tym podrozdziale zostanie wyznaczona optymalna sieć neuronowa MLP bez preprocessingu (przetwarzania wstępnego) danych wejściowych, czyli zawierająca wszystkie 56 zmiennych wejściowych. Najlepszą siecią okazał się model MLP 56:6:1, czyli z 6 neuronami w warstwie ukrytej. Poniższa tabela przedstawia wyniki uzyskane dla sieci bez preprocessingu danych wejściowych. I tak kolejno pierwsza kolumna przedstawia statystyki regresyjne dla najlepszej uzyskanej sieci MLP 56:6:1 przy prognozie na 1 okres naprzód, druga dla średniej z dziesięciu sieci neuronowych, przy czym w tym przypadku ilość neuronów w warstwie ukrytej wahała się między 6 a 13. Trzecia kolumna przedstawia statystyki regresyjne dla najlepszej sieci przy prognozie 2 okresy naprzód. W tym przypadku najlepszą siecią okazała się sieć MLP 56:5:1 z 5 neuronami ukrytymi.

	<i>Najlepsza sieć dla t+1 56:6:1</i>			<i>Średnia z 10 sieci 56:6-13:1</i>			<i>Najlepsza sieć dla t+2 56:5:1</i>		
	<i>Tr</i>	<i>V</i>	<i>Te</i>	<i>Tr</i>	<i>Ve</i>	<i>Te</i>	<i>Tr</i>	<i>V</i>	<i>Te</i>
<i>Odch. błędu</i>	27.197	27.019	27.818	27.858	27.950	29.908	39.483	38.446	38.499
<i>Śr. bł. bezwz.</i>	19.846	19.880	20.625	20.439	20.647	22.136	29.025	29.725	29.441
<i>RMS</i>	0.0182	0.0181	0.0186	0.0187	0.0187	0.0201	0.0261	0.0255	0.0255
<i>SD ratio</i>	0.085	0.094	0.069	0.088	0.097	0.074	0.124	0.134	0.095
I^2	0.00029	0.00029	0.00026	0.00030	0.00031	0.00031	0.0006	0.00059	0.00051
<i>I</i>	1.697%	1.711%	1.623%	1.739%	1.770%	1.753%	2.463%	2.439%	2.249%
I_1^2	0.269%	0.007%	2.041%	0.353%	0.172%	5.743%	0.271%	0.037%	1.297%
I_2^2	0.029%	0.029%	0.026%	0.030%	0.031%	0.029%	0.061%	0.059%	0.050%
I_3^2	99.730%	99.829%	97.947%	99.605%	99.689%	93.429%	99.729%	99.611%	98.242%
<i>U</i>	0.924			0.959			1.324		

Tabela 1. Statystyki dla sieci neuronowych bez preprocessingu danych wejściowych.

Aby móc zrozumieć znaczenie poszczególnych parametrów regresyjnych poniżej została przedstawiona ich krótka charakterystyka.

Odchylenie błędu - Odchylenie standardowe błędu dla zmiennej wyjściowej;

Średni błąd bezwzględny – Średni błąd rozumiany jako moduł różnicy pomiędzy wartością zadaną i wartością uzyskaną na wyjściu;

RMS – Średniokwadratowy błąd prognozy;

S.D. Ratio. - Iloraz odchylenia standardowego błędów i odchylenia standardowego wartości zmiennej objaśnianej. Miernik ten przyjmuje zawsze wartości nieujemne. Mniejsza jego wartość świadczy o lepszej jakości modelu. Dla bardzo dobrego modelu miernik ten osiąga wartość z przedziału od 0 do 0,1. Jeśli wartość *S.D. Ratio* jest większa od jedności, to korzystanie z tak skonstruowanego modelu jest nieuzasadnione, gdyż dokładniejszym oszacowaniem wartości zmiennej objaśnianej jest jej średnia arytmetyczna wyznaczona na podstawie zbioru uczącego.

Większość badaczy poprzestaje tylko na powyższych parametrach dodając jeszcze jeden – współczynnik korelacji Pearsona pomiędzy wartością zadaną, a wartością uzyskaną na wyjściu.

Gdyby na tym poprzestać model wydawałby się wręcz idealny, gdyż średniokwadratowy błąd prognozy dla $t+1$ wynosi poniżej 2% (dla $t+2$ wynosi około 2,6%), średni błąd bezwzględny informujący nas o ile średnio odpowiedzi sieci różnią się od rzeczywistych wartości prognozowanych wahają się w zależności od modelu pomiędzy 20, a 30. Jednak jak się okazuje budowanie zyskowego modelu decyzyjnego na podstawie prognozy na 2 okresy naprzód jest bezsensowne. Dzieje się tak za sprawą interpretacji kolejnych dwóch parametrów regresyjnych.

Współczynnik I^2 Theila:

$$I^2 = \frac{\sum_{t=1}^n (a_t - p_t)^2}{\sum_{t=1}^n a_t^2} \quad (3.8)$$

gdzie: a_t to szereg wartości rzeczywistych, a p_t to szereg wartości prognozowanych.

Współczynnik I^2 Theila jest stosunkowo mało znany, a bardzo użyteczny przy ocenie działania sieci neuronowych w analizie szeregów czasowych. Współczynnik ten przyjmuje wartość równą zero w sytuacji, gdy wyznaczone prognozy są całkowicie zgodne z wartościami rzeczywistymi. Mogłoby się zatem wydawać, że wartości współczynnika I^2 z powyższej tabeli są do pominięcia. Gdy jednak wyliczymy pierwiastek z I^2 , który informuje nas o przeciętnym względnym błędzie prognozy w badanym okresie, okaże się że

prognozy dalekie są od ideału, biorąc pod uwagę, że są to prognozy na 1 i 2 okresy do przodu, a zmiany indeksu podczas jednej sesji rzadko przekraczają 1,5 do 2%.

Przeciętny względny błąd prognozy nie informuje nas jednak co jest przyczyną zaistniałego błędu. Chcąc określić przyczynę generowania nieprawidłowych prognoz można rozłożyć współczynnik I^2 Theila na trzy składniki:

$$I^2 = I_1^2 + I_2^2 + I_3^2 \quad (3.9)$$

Powyższe składniki definiowane są następująco:

$$I_1^2 = \frac{(\bar{a} - \bar{p})^2}{\frac{1}{n} \sum_{t=1}^n a_t^2} \quad (3.10)$$

$$I_2^2 = \frac{(s_a - s_p)^2}{\frac{1}{n} \sum_{t=1}^n a_t^2} \quad (3.11)$$

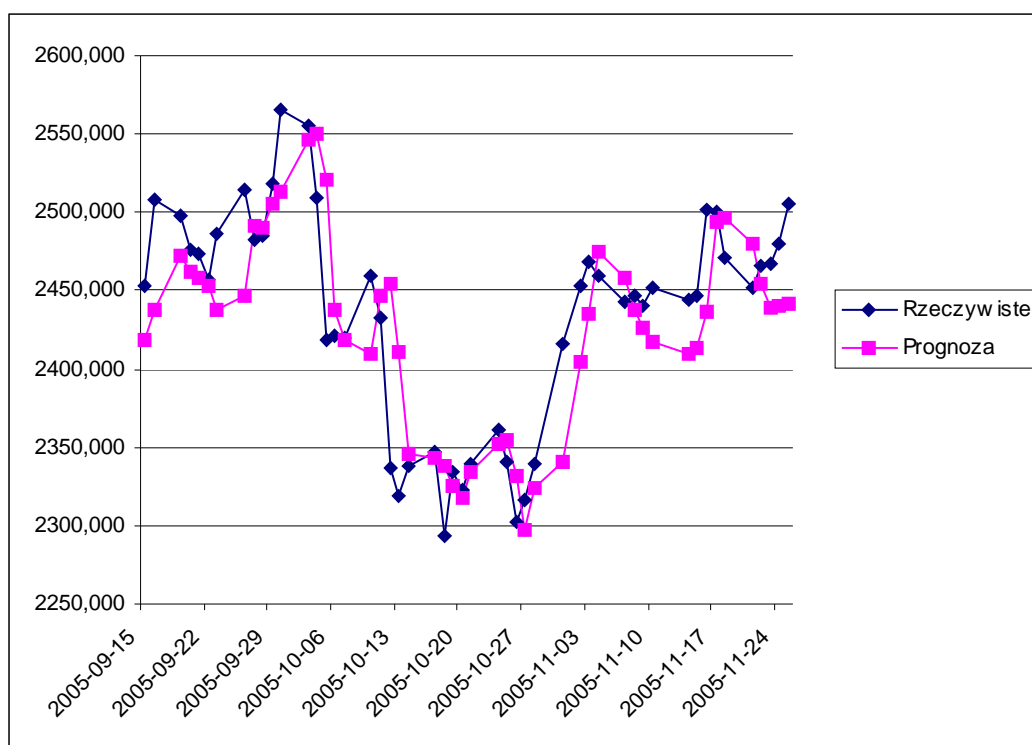
$$I_3^2 = \frac{2s_a s_p (1-r)}{\frac{1}{n} \sum_{t=1}^n a_t^2} \quad (3.12)$$

gdzie: \bar{a} i \bar{p} to odpowiednio wartości średnie dla szeregu wartości rzeczywistych i prognozowanych, s_a jest odchyleniem standardowym dla szeregu wartości rzeczywistych, s_p jest odchyleniem standardowym dla wartości prognozowanych, r jest współczynnikiem korelacji liniowej pomiędzy wartościami a_t i p_t .

Wartość I_1^2 reprezentuje miarę obciążenia prognozy, I_2^2 mierzy błąd spowodowany niedostateczną elastycznością prognozy, czyli brakiem dokładności w przewidywaniu wahań zmiennej prognozowanej, natomiast I_3^2 informuje o błędzie związanym z niedostateczną zgodnością kierunku zmian prognoz w porównaniu z kierunkiem zmian wartości prognozowanej. Zatem prawie wszystkie błędy w powyższych modelach wynikają właśnie z braku zgodności kierunku zmian prognoz z kierunkiem zmian wartości prognozowanej, co dobrze obrazują rzeczywiste i prognozowane wartości indeksu zawarte w tabeli 2. Widać z nich bowiem, że tak zaprojektowana sieć neuronowa nie przewiduje gwałtowniejszych zmian wartości indeksu, a w szczególności kierunku tych zmian.

Data	Rzeczywiste	Prognozowane
2005-09-22	2456.56	2453.08
2005-09-23	2486.45	2437.44
2005-09-26	2514.84	2446.60
2005-09-27	2482.69	2491.93
2005-09-28	2484.42	2490.66
2005-09-29	2518.73	2505.59
2005-09-30	2564.88	2512.87
2005-10-03	2554.79	2546.40
2005-10-04	2508.90	2550.42
2005-10-05	2418.65	2520.47
2005-10-06	2421.12	2437.33
2005-10-07	2419.70	2418.74
2005-10-10	2460.08	2409.82
2005-10-11	2432.18	2446.28
2005-10-12	2336.38	2453.86
2005-10-13	2318.77	2410.54

Tabela 2. Przykładowe zestawienie rzeczywistych i prognozowanych wartości indeksu WIG20.



Rys. 5. Rzeczywiste i prognozowane wartości indeksu WIG20 w okresie 15-09-05 do 25-11-05.

W tabeli 1 zamieszczony jest jeszcze jeden parametr, będący względną miarą poprawności modelu umożliwiającą porównanie jakości prognoz z prognozami otrzymanymi przy pomocy stanowiącego punkt odniesienia pewnego modelu wzorcowego. Jest to statystyka *U Theila* definiowana wzorem:

$$U = \frac{\sqrt{\sum_{t=1}^n (a_t - p_t)^2}}{\sqrt{\sum_{t=1}^n (a_t - a_{t-1})^2}} \quad (3.13)$$

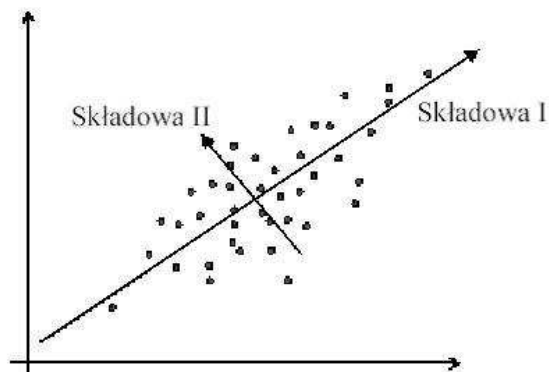
Celem tej statystyki jest porównanie rezultatów uzyskanych przy pomocy badanego modelu z tak zwaną prognozą naiwną, w której zakłada się, że prognoza na okres $t+1$ jest równa wartości szeregu w chwili t . Jeśli statystyka *U Theila* przyjmuje wartości powyżej jedności wówczas prognoza naiwna jest lepsza od prognozy uzyskanej przy pomocy konstruowanych modeli. Widać zatem, że uzyskany model prognostyczny dla $t+1$ może służyć do prognozowania przyszłej wartości indeksu WIG20 (*U Theila* dla najlepszej sieci wynosi 0,924), natomiast w przypadku modelu dla $t+2$ prognoza naiwna jest lepsza od prognozy uzyskanej przy pomocy prognostycznego modelu neuronowego.

W dalszych podrozdziałach przedstawione zostaną różne dostępne w pakiecie Statistica Neural Networks metody optymalizacji wektora zmiennych wejściowych i ich wpływ na jakość modeli prognostycznych.

3.3.2. Analiza głównych składowych (Principal Components Analysis).

Najpopularniejszym sposobem redukcji wymiaru przestrzeni sygnałów wejściowych jest analiza głównych składowych [7]. Jest to liniowa technika w teorii sygnałów znana jako transformata Kerhunena-Loeve'a. Jest też bardzo bliska metodzie najmniejszych kwadratów, analizie czynnikowej, metodzie rozkładu macierzy wg wartości szczególnych i filtracji dopasowanej. Celem tej metody jest znalezienie M ortogonalnych wektorów w przestrzeni danych, których wpływ jest największy na wariancję danych. Rzut danych z przestrzeni N – wymiarowej na M – wymiarową podprzestrzeń rozpiętą przez te wektory, umożliwia redukcję wymiarowości, przy zachowaniu większości istotnych informacji zawartych w zbiorze danych. Na ogół $M \ll N$.

Ścisłej mówiąc przyjmuje się, że pierwsza główna składowa leży wzdłuż kierunku maksymalnej wariancji. Druga główna składowa ma kierunek maksymalnej wariancji w podprzestrzeni prostopadłej do pierwszej składowej. Trzecia główna składowa ma kierunek maksymalnej wariancji w podprzestrzeni prostopadłej do dwóch pierwszych, itd. Ilustrację graficzną metody głównych składowych przedstawia rysunek 6.



Rys. 6. Graficzna ilustracja metody PCA (kierunki składowych w dwóch wymiarach).

Ogólnie wykazać można, że kierunek k -tej głównej składowej pokrywa się z kierunkiem wektora własnego, odpowiadającego k -tej co do wielkości wartości własnej, pełnej macierzy kowariancji [51]. Dla danych o zerowej średniej wszystko sprowadza się do odpowiednich wektorów własnych macierzy autokorelacji A :

$$A = \sum_p (x_p - \bar{x})^T (x_p - \bar{x}) \quad (3.14)$$

gdzie x_p jest p -tym elementem zbioru uczącego.

Niedogodnością używania tej metody są problemy z interpretacją ładunków czynnikowych (określających położenie punktów w nowo wygenerowanym układzie współrzędnych) przy większej liczbie pierwotnych zmiennych. Czynniki tworzone tą metodą są liniową kombinacją tych zmiennych pierwotnych, które są skorelowane najmocniej z powstałym czynnikiem. Między innymi stąd im więcej zmiennych pierwotnych, tym trudniejsza interpretacja „ładunków czynnikowych“.

Pomimo wspomnianej niedogodności metoda ta jest często stosowana, ponieważ pozwala wyodrębnić z pierwotnych danych wejściowych bardzo małą liczbę głównych składowych o tak dużej zawartości informacyjnej, że podanie ich na wejście sieci modelującej badaną zależność z powodzeniem może zastąpić podawanie wszystkich oryginalnych danych wejściowych. Niestety metoda ta ma jeszcze jedną wadę. W przypadku szerokiego stosowania PCA, pewnym ograniczeniem może być liniowy charakter tej techniki, który w niektórych przypadkach może powodować utratę ważnych informacji na temat struktury danych (dane w trakcie „rzutowania“ mogą się wzajemnie „przesłaniać“).

Pakiet Statistica umożliwia ocenę stopnia zachowania informacji w dowolnie ograniczonej podprzestrzeni poprzez skumulowany współczynnik informacji, który pokazuje stopień zachowania pierwotnej wariancji zbioru M -wymiarowego przez podzbiór N -wymiarowy. Aby zachować 99% informacji zawartych w 56-wymiarowej przestrzeni wejściowej przyjętego modelu, należy ograniczyć przestrzeń wejść do 16 wymiarów (tabela 3.), a dla zachowania 99,95% przestrzeń wejść należy ograniczyć do 24 wymiarów. Z danych zawartych w tabeli 3 wynika ponadto zdecydowana dominacja pierwszych pięciu głównych składowych, na które przypada prawie 90% zmienności, a 40 końcowych głównych składowych to mniej niż 1% zmienności.

Składowe główne (dane wejściowe)			
	Wartość własna	Informacja	Skumulowana
1	1424,561	0,569683	0,569683
2	433,208	0,173240	0,742924
3	184,455	0,073764	0,816688
4	136,642	0,054643	0,871331
5	61,345	0,024532	0,895863
6	47,643	0,019053	0,914915
7	42,505	0,016998	0,931913
8	31,510	0,012601	0,944514
9	28,035	0,011211	0,955726
10	23,318	0,009325	0,965051
11	19,664	0,007864	0,972914
12	13,329	0,005330	0,978245
13	10,617	0,004246	0,982491
14	9,749	0,003899	0,986389
15	7,959	0,003183	0,989572
16	7,560	0,003023	0,992595
17	3,661	0,001464	0,994059
18	3,468	0,001387	0,995446
19	2,941	0,001176	0,996622
20	2,528	0,001011	0,997634
21	2,292	0,000917	0,998550
22	1,650	0,000660	0,999210
23	0,531	0,000212	0,999422
24	0,315	0,000126	0,999548

Tabela 3. Stopień zachowania pierwotnej wariancji przestrzeni wejść.

Najlepszą siecią do realizacji preprocessingu metodą analizy głównych składowych, okazał się model MLP 16:6:1 oraz 24:7:1. Wskaźniki regresyjne dla tych sieci przedstawia tabela 4.

	Najlepsza sieć dla 16:6			Najlepsza sieć dla 24:7			Średnia z 10 sieci 24:7		
	<i>Tr</i>	<i>V</i>	<i>Te</i>	<i>Tr</i>	<i>Ve</i>	<i>T</i>	<i>Tr</i>	<i>V</i>	<i>Te</i>
<i>Odch. Błędu</i>	29.830	29.875	30.494	29.458	29.262	29.060	29.919	29.344	30.885
<i>Śr. bł. bezwz.</i>	21.516	21.765	22.339	21.628	21.631	21.530	22.054	21.799	22.811
<i>RMS</i>	0.0199	0.02	0.0205	0.0197	0.0195	0.0195	0.0199	0.0194	0.0206
<i>SD ratio</i>	0.094	0.104	0.075	0.093	0.102	0.072	0.094	0.102	0.076
I^2	0.00035	0.00036	0.00032	0.00034	0.00034	0.00029	0.00035	0.00034	0.00031
<i>I</i>	1.862%	1.891%	1.786%	1.839%	1.852%	1.695%	1.861%	1.843%	1.748%
I_1^2	0.000%	0.000%	0.816%	0.000%	0.037%	0.011%	0.004%	0.031%	0.192%
I_2^2	0.218%	0.053%	0.525%	0.610%	0.089%	1.604%	0.597%	0.056%	2.322%
I_3^2	99.782%	99.947%	98.659%	99.389%	99.874%	98.385%	99.399%	99.913%	97.486%
<i>U</i>	1.016			0.992			1.006		

Tabela 4. Statystyki dla sieci neuronowych z preprocessingiem danych w postaci Analizy Głównych Składowych.

Z tabeli 4 wynika, że zastosowanie preprocessingu danych wejściowych w postaci analizy głównych składowych przyniosło nieoczekiwany znaczący wzrost wartości błędów. Szczególnie widoczne jest to po statystyce *U* Theila wahającej się w okolicach jedności, co sugeruje, że równie „dobrą” prognozą wartości indeksu co prognostyczny model neuronowy byłaby prognoza naiwna, czyli wartość indeksu z poprzedniego okresu. Może być to spowodowane wspomnianą wcześniej niedogodnością tej metody czyli jej liniowym charakterem, który mógł spowodować utratę pewnych ważnych informacji na temat struktury danych.

3.3.3. Wyznaczenie optymalnego wektora wejściowego przy pomocy analizy wrażliwości.

Analiza wrażliwości wejścia sieci neuronowej pozwala na odróżnienie ważnych zmiennych od takich, które niewiele wnoszą do wyniku działania sieci, te ostatnie można odrzucić. Analizę wrażliwości można wykonywać tylko po to by dowiedzieć się czegoś o zmiennych wejściowych, albo w celu odrzucenia zbędnych zmiennych. Analiza wrażliwości daje wgląd w użyteczność poszczególnych zmiennych wejściowych. Wskazuje zmienne, które, bez straty jakości sieci mogą być pominięte i zmienne kluczowe, których nigdy nie wolno pomijać. Z pewnych względów, opisanych niżej, konieczna jest jednak ostrożność przy wyciąganiu wniosków.

Zmienne wejściowe, z reguły nie są niezależne. Analiza wrażliwości wykazuje stratę jaką ponosimy odrzucając konkretną zmienną. Jednak wobec zależności pomiędzy

zmiennymi, taki wskaźnik obliczany niezależnie, dla każdej zmiennej może nie odzwierciedlać rzeczywistej sytuacji.

Założmy, na przykład, że dwie zmienne zawierają tę samą informację, a nawet, że są po prostu kopiami. Usunięcie jednej ze zmiennych nie powoduje straty, gdyż informacja dostępna jest w drugiej. Tak więc, obie te zmienne będą, w niezależnych krokach uznane za nieważne. Podczas gdy jednoczesne usunięcie obu zmiennych pozbawi się bardzo ważnej informacji.

Zdarzają się też takie, powiązane wzajemnie zmienne, które użyteczne są tylko razem. Gdy włączone są wszystkie te zmienne, to każda z nich da dużą wrażliwość. Jednak, gdy nie wszystkie są uwzględnione, to wrażliwość na pozostałe będzie zerowa. Reasumując, analiza wrażliwości nie daje bezwzględnej oceny użyteczności zmiennych. Trzeba ją stosować ostrożnie, co nie zmienia faktu jej wielkiej, praktycznej przydatności. Badając więcej modeli znajdujemy powtarzające się, ważne zmienne i powtarzające się pomijalne, oraz takie, które czasami dają dużą wrażliwość a czasami małą, te ostatnie zawierają zapewne redundantną informację.

W tabeli 5 podana jest wrażliwość sieci na utratę dziesięciu najistotniejszych według rangi zmiennych. Tabela 6 przedstawia natomiast zmienne, dla których wartość ilorazu jest mniejsza od 1, czyli te zmienne, których usunięcie powinno poprawić jakość modelu.

	MA 9	BUX-1	WIG20 L	WIG20 H	SP500O	WIG20C-1	SP500C	WIG20 C	BUX O	Nasdaq C
Iloraz	3,3169	3,1485	2,9925	2,9271	2,8598	2,7338	2,1797	1,9503	1,8712	1,6656
Ranga	1	2	3	4	5	6	7	8	9	10

Tabela 5. Rangi 10 najistotniejszych wejść wg Analizy Wrażliwości.

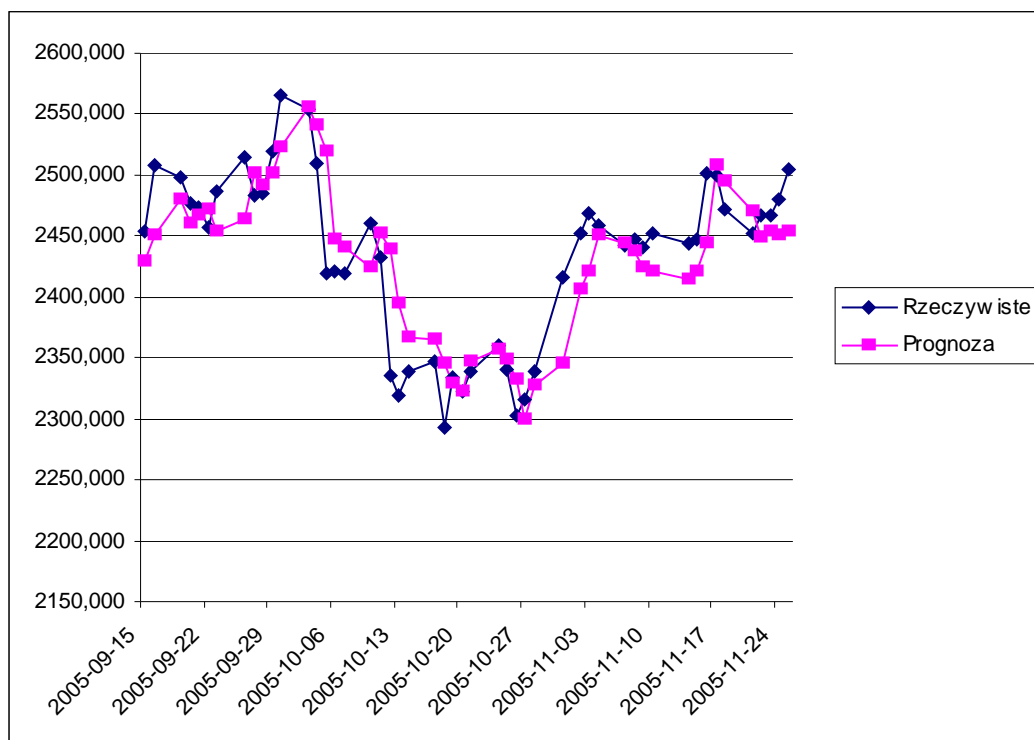
	ZMN57	RSI 9	RSI 14	ZMN52	ZMN51	ŚredniaV	ATR 14	WIG20-2	Volume	Nasdaq-2	EMA 5
Iloraz	0,9999	0,9998	0,9992	0,9988	0,9986	0,9970	0,9939	0,9933	0,9920	0,9909	0,9872
Ranga	46	47	48	49	50	51	52	53	54	55	56

Tabela 6. Zestawienie zmiennych wejściowych negatywnie wpływających (wg analizy wrażliwości) na jakość modelu.

Przestrzeń wejść ograniczona zostaje zatem do 45 wymiarowej podprzeźreni, gdyż wartości ilorazu dla pozostałych zmiennych objaśniających były mniejsze od 1. Najlepszą siecią uczoną tak przetworzonym sygnałem wejściowym okazał się model MLP 45:7:1. wskaźniki regresyjne dla tej sieci przedstawia tabela 7.

	Najlepsza sieć dla 45:7		
	Tr	V	Te
Odch. Błędu	26.774	27.352	27.804
Śr. bł. bezwz.	19.638	20.126	20.504
RMS	0.0179	0.0183	0.0186
SD ratio	0.084	0.095	0.069
I^2	0.00028	0.00030	0.00026
I	1.671%	1.733	1.622%
I_1^2	0.000%	0.200%	0.006%
I_2^2	0.199%	0.011%	1.085%
I_3^2	99.800%	99.789%	98.909%
U	0.918		

Tabela 7. Statystyki regresyjne dla sieci MLP 45:7:1.



Rys. 7. Rzeczywiste i prognozowane wartości indeksu WIG20 w okresie 15-09-05 do 25-11-05.

Z przedstawionych w tabeli 7 statystyk regresyjnych dla sieci neuronowej z preprocessingiem danych wejściowych w postaci Analizy Wrażliwości, można zauważyć, iż daje ona nieznacznie lepsze wyniki prognostyczne od sieci neuronowej tworzonej przy użyciu wszystkich 56 zmiennych. Niestety niezmiernie kłopotliwe przy tej metodzie było wyszukanie optymalnej sieci i ustalenie nieistotnych zmiennych objaśniających, ponieważ

występowała duża zmienność zarówno błędów regresyjnych jak i usuwanych zmiennych. Podobny problem występował również przy kolejnej metodzie dostępnej w pakiecie Statistica – Automatycznego projektanta sieci.

3.3.4. Wyznaczenie optymalnego wektora wejściowego przy pomocy Automatycznego Projektanta Sieci.

Najwygodniejszym sposobem redukcji wektora danych wejściowych przy wykorzystaniu pakietu SNN jest skorzystanie z automatycznego projektanta sieci. Wyřęca on użytkownika zarówno w procesie budowy modelu jak też w odniesieniu do optymalizacji podzbioru wejściowego. Firma StatSoft nie prezentuje jednak szczegółów jego działania. Można przypuszczać jednak, że jest to metoda hybrydowa, oparta na takich technikach jak: algorytmy genetyczne, analiza wrażliwości i selekcja krokowa.

Przy wykorzystaniu tego narzędzia, podobnie jak w poprzednim przypadku, mamy do czynienia z dużą zmiennością tworzonych sieci i wyboru danych wejściowych, a co za tym idzie kłopotliwe jest wyszukanie optymalnej architektury sieci. Ten problem dobrze przedstawia tabela 8, obrazująca jakie zmienne wejściowe zostały dobierane do poszczególnych modeli. Widać, że ilość zmiennych dobieranych przez Automatycznego Projektanta waha się od 5 do 32. Takie „niezdecydowanie” ma również swoje przełożenie na średnią jakość budowanych modeli, która jest gorsza od średniej jakości modeli bez preprocessingu danych wejściowych.

Optymalnym modelem okazał się model MLP 30:8:1, którego statystyki regresyjne są nieznacznie lepsze od tych uzyskanych przy modelu z wykorzystaniem całego 56-elementowego wektora wejściowego.

<i>Wariant 1</i>	<i>Wariant 2</i>	<i>Wariant 3</i>	<i>Wariant 4</i>	<i>Wariant 5</i>
WIG20-3	Wig20 C	WIG20-3	WIG20 H	WIG20 L
WIG20-1	Volume	WIG20-2	WIG20 L	WIG20C-3
WIG20 O	średniaV	WIG20-1	CCI 14	Volume
WIG20 H	bux-1	WIG20 H	SP500 O	CCI 14
WIG20C-2	ZMN60	Volume	Nasdaq O	EMA 9
WIG20 C		CCI 9	BUX C	SP500 C
Volume		CCI 14	ZMN47	Nasdaq-1
CCI 14		ATR 14	ZMN60	Nasdaq C
RSI 9		RSI 14		DAX O
RSI 14		ROC 10		DAX-1
ROC 5		ŚredniaV		USD
ROC 10		EMA 5		ZMN52
ŚredniaV		EMA 12		ZMN60
EMA 5		MA 12		
MA 9		WIG O		
MA 12		SP500 O		
WIG O		Nasdaq O		
WIG H		Nasdaq-1		
SP500 C		Nasdaq C		
Nasdaq O		DAX-1		
Nasdaq-1		DAX C		
Nasdaq C		BUX O		
DAX C		BUX C		
BUX C		USD		
ZMN48		ZMN47		
ZMN49		ZMN51		
ZMN50		ZMN52		
ZMN54		ZMN54		
ZMN57		ZMN56		
ZMN59		ZMN57		
ZMN60		ZMN59		
		ZMN60		

Tabela 8. Dobór zmiennych wejściowych przy pomocy Automatycznego Projektanta.

	Najlepsza sieć dla 30:8:1			Średnia 10 sieci 54:3-7:1		
	Tr	V	Te	Tr	Ve	T
Odch. Błędu	26.670	26.728	28.747	28.137	28.401	29.584
Śr. bł. bezwz.	19.534	19.740	21.382	19.807	20.222	21.910
RMS	0.0178	0.0179	0.0193	0.0188	0.0190	0.0198
SD ratio	0.084	0.093	0.071	0.088	0.099	0.073
I^2	0.00028	0.00029	0.00028	0.00031	0.00032	0.00030
I	1.664%	1.693%	1.678%	1.756%	1.798%	1.729%
I_1^2	0.000%	0.168%	0.090%	0.002%	0.080%	0.381%
I_2^2	0.188%	0.054%	1.971%	0.215%	0.014%	3.146%
I_3^2	99.812%	99.777%	97.939%	99.783%	99.906%	96.473%
U	0.918			0.965		

Tabela 9. Statystyki regresyjne dla sieci MLP 30:8:1.

3.3.5. Dobór zmiennych wejściowych przy pomocy algorytmu genetycznego.

Idea algorytmów genetycznych jest inspirowana procesami, obserwowanymi w przyrodzie, którymi są selekcja osobników i ewolucja gatunków, mechanizmy rozmnażania i związane z nimi dziedziczenie cech. W wyniku działania mechanizmów naturalnych powstają nowe gatunki, lepiej dostosowane do środowiska, w którym żyją, wypierając gatunki gorzej dostosowane. Opis zjawisk związanych z doborem naturalnym i dziedziczeniem cech okazał się bardzo inspirujący i przydatny do optymalizacji wielu różnorodnych układów złożonych. Uniwersalność tej metody wynika z przyjętego założenia, że nie operują one na rzeczywistych danych, a tylko na odpowiednio zakodowanych ich reprezentacjach. Sposób kodowania może w istotnej mierze wpłynąć na skuteczność tej metody.

W analizowanym przypadku zbiór danych wejściowych zostanie zakodowany jako 56-elementowy ciąg binarny. Wystąpienie jedynek na danej pozycji oznacza, że odpowiednia dla niej zmienna objaśniająca, powinna być reprezentowana w podzbiorze uczącym. Obecność zera eliminuje ją ze zbioru. Powstaje więc struktura w postaci łańcucha cech, czyli występuje analogia do chromosomu i zawartych w nim genów. Aby w pełni przeanalizować problem, należałoby sprawdzić jakość uczenia sieci przy wszystkich możliwych permutacjach tego łańcucha, co z oczywistych względów nie jest możliwe. Poszukiwanie najlepszych rozwiązań w sposób losowy jest dobrym pomysłem, ale dopiero wtedy, gdy postaramy się chociaż trochę dopomóc „ślepeму losowi”.

Na tym właśnie opiera się idea algorytmu genetycznego. Tworzy on w sposób losowy populację takich łańcuchów, a następnie realizuje proces analogiczny do naturalnej selekcji,

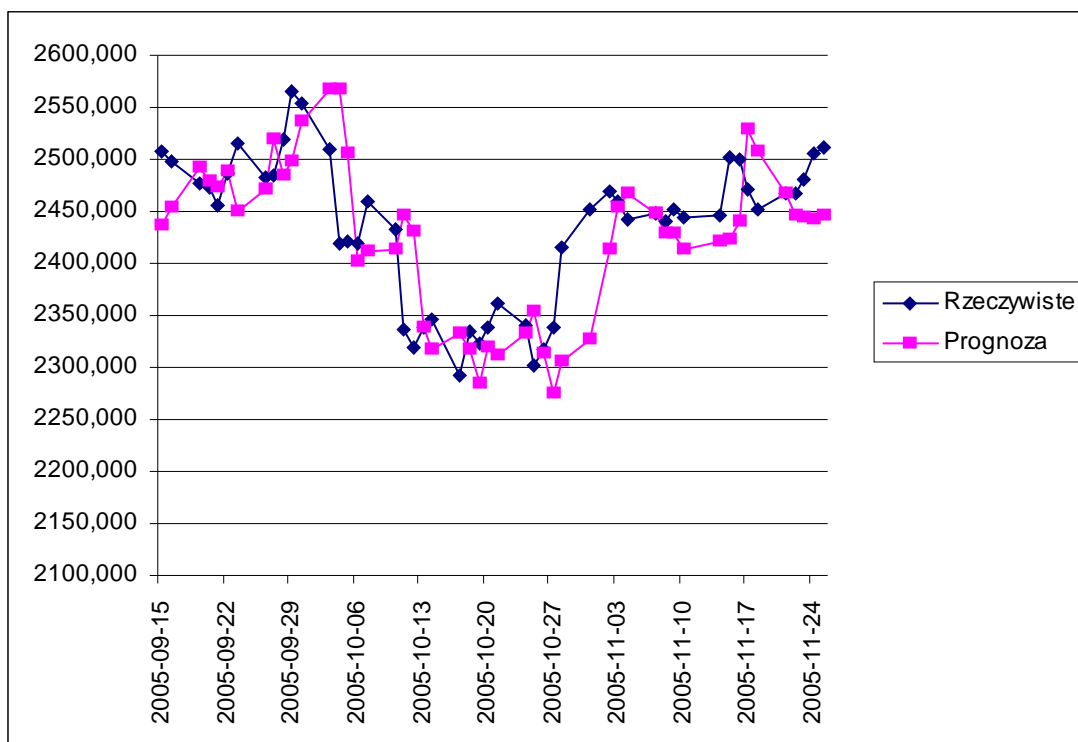
zachodzącej w przyrodzie podczas ewolucji. Celem tego modelowanego w komputerze procesu, jest wybór lepszych łańcuchów, odpowiadających najlepszym sieciom. Te właśnie łańcuchy są punktem wyjścia do utworzenia nowej populacji „osobników potomnych”. Osobnikiem potomnym w algorytmie genetycznym jest łańcuch binarny wytworzony na podstawie jednego lub dwóch binarnych łańcuchów rodzicielskich (w zależności od tego czy jest to rozmnażanie bezpłciowe – jeden łańcuch, czy płciowe – dwa łańcuchy rodzicielskie). Przy tworzeniu potomka łańcuchy rodzicielskie podlegają pewnym losowym zmianom, określanym jako mutacje i krzyżowanie. Powodują one to, że osobniki potomne posiadają większość cech swoich rodziców, lecz mają także nieco zmodyfikowane właściwości. Sprawia to, że już po kilku, kilkunastu pokoleniach można zaobserwować powstawanie łańcuchów, które ze względu na tzw. funkcję celu wykazują się lepszym przystosowaniem. Funkcją celu w rozważanym przypadku jest iloraz odchyleń, obliczany dla zbioru walidacyjnego. Pewna losowość tej metody pod postacią współczynników krzyżowania i mutacji sprawia, że należy liczyć się z okresowym pogarszaniem się wyników. Zazwyczaj przy odpowiednio dobranych parametrach algorytm genetyczny przynosi dobre rezultaty. Ta metoda doboru zmiennych wejściowych ma jedną zasadniczą zaletę. W porównaniu z poprzednimi - dobrze sobie radzi ze zmiennymi wzajemnie uzależnionymi (wymagającymi wspólnej obecności) [28]. Jednak zmienne takie powinny być zlokalizowane blisko siebie w łańcuchu tworzącym maskę. Wynika z tego, że jeśli mamy powody, aby oczekiwać, że jakieś konkretne zmienne są mocno ze sobą powiązane, to powinniśmy umieścić je w sąsiednich kolumnach arkusza danych. Spowoduje to, że ich współzależności będą szczególnie pilnie badane przez algorytm genetycznego doboru zmiennych wejściowych.

Dobór cech w analizowanym przypadku realizowany był na populacji 100 elementowej przez 100 pokoleń. Wartość współczynnika mutacji została ustalona na 0,01, współczynnika krzyżowania na 0,9, a kary jednostkowej⁴ (kara za cechę) na 0. Algorytm genetyczny ograniczył przestrzeń wejść do zbioru 54-elementowego czyli zostały usunięte zaledwie 2 zmienne: ATR14 (średnia rzeczywistego zasięgu) i dynamikę zmiany logarytmicznych dwudniowych stóp zwrotu. Przyjęcie współczynnika kary za cechę na poziomie różnym od 0 sprawiało, że Algorytm genetyczny usuwał wszystkie zmienne wejściowe.

⁴ Współczynnik ten przemnażany jest przez liczbę wybranych w każdej masce zmiennych wejściowych, a następnie dodawany do błędu walidacyjnego sieci w trakcie jej uczenia i testowania. Niezerowa wartość kary za cechę faworyzuje mniejsze sieci, i zwykle zwiększa sprawność. Jednakże, gdy współczynnik ten jest zbyt duży, to liczba zmiennych staje się ważniejsza od jakości sieci. Zbyt duża wartość kary za cechę może nawet spowodować, że algorytm usunie wszystkie zmienne wejściowe.

	Najlepsza sieć dla 54:5:1			Średnia 10 sieci 54:3-7:1		
	Tr	V	Te	Tr	Ve	T
Odch. Błędu	26.848	26.855	28.020	26.900	27.389	29.589
Śr. bł. Bezwz.	19.806	19.793	20.834	19.807	20.222	21.910
RMS	0.018	0.018	0.0188	0.0180	0.0184	0.0199
SD ratio	0.084	0.094	0.069	0.085	0.096	0.073
I^2	0.00028	0.00029	0.00027	0.00028	0.00030	0.00030
I	1.676%	1.702%	1.636%	1.679%	1.736%	1.732%
I_1^2	0.007%	0.215%	0.132%	0.003%	0.240%	0.640%
I_2^2	0.216%	0.001%	2.766%	0.257%	0.030%	6.021%
I_3^2	99.777%	99.784%	97.102%	99.739%	99.729%	93.339%
U	0.918			0.934		

Tabela 10. Statystyki regresyjne dla sieci uzyskanej przy pomocy Algorytmu Genetycznego.



Rys. 8. Rzeczywiste i prognozowane wartości indeksu WIG20 w okresie 15-09-05 do 25-11-05.

W niniejszym przypadku optymalną siecią okazał się model MLP 54:5:1, a obserwując statystyki regresyjne zamieszczone w tabeli 10, można zauważyć, że są one zbliżone do tych uzyskanych zarówno przy pomocy automatycznego projektanta jak i analizy wrażliwości. Jest jednak różnica, która dzieli te modele, a mianowicie zmienność pod postacią średniej z wielu różnych sieci powstałych w procesie uczenia. Ta zmienność jest najmniejsza dla sieci powstałych właśnie przy użyciu preprocessingu danych w postaci Algorytmu

Genetycznego. Można zatem domniemywać, że usunięte tylko 2 zmienne wejściowe w istotny sposób zredukowały niekorzystny szum występujący w danych.

3.3.6. Wyznaczenie optymalnego wektora wejściowego przy pomocy nieliniowej kompresji danych.

Omawiając poprzednio metodę analizy głównych składowych wspomniana została pewna wada tej metody, mianowicie liniowy charakter tej techniki, który w niektórych przypadkach może powodować utratę ważnych informacji na temat struktury danych. Tej niedogodności pozbawiona jest metoda nieliniowej kompresji danych przy pomocy sieci autoasocjacyjnej [7], [10], [23]. Jest to sieć, która jest uczona w taki sposób, aby dokonywała maksymalnie wiernej reprodukcji wartości wejściowej na swoich wyjściach. Na pozór takie działanie nie ma sensu, bo skoro mamy pewne dane na wejściu, to po co budować sieć i uczyć ją, żeby odtworzyła na swoim wyjściu dokładnie takie same dane. Jednak działanie takie zdecydowanie ma sens, ponieważ rozważana sieć autoasocjacyjna posiada w warstwie ukrytej zdecydowanie mniejszą liczbę neuronów niż w warstwie wejściowej czy wyjściowej. Informacja zatem przechowywana jest w sposób rozproszony w postaci odpowiednich wartości wag sieci. Stąd nazwa sieci, której działanie przypomina ludzką pamięć skojarzeniową. Pojemność sieci ogranicza liczba neuronów w warstwie ukrytej [31]. W przypadku prezentowania wzorców rzeczywistych, zmniejszanie liczby tych neuronów, skutkować będzie wzrostem błędu na wyjściu sieci. Z kolei sieć o mniejszej liczbie neuronów w warstwie ukrytej, powinna mieć większą zdolność do eliminacji szumów zawartych w sygnale wejściowym i jednocześnie powinna charakteryzować się większą zdolnością do generalizacji. Jeśli wyobrazimy sobie sieć MLP 9:3:9 to odtworzenie przybliżonych danych wejściowych na wyjściu sieci powoduje, że sieć musi dokonać kompresji sygnału 9-wymiarowego do 3-wymiarowego.

Dlatego też w celu realizacji stawianego jej zadania reprodukcji informacji wejściowej na wyjściu sieć musi się najpierw nauczyć reprezentacji obszernych danych wejściowych przy pomocy mniejszej liczby sygnałów produkowanych przez neurony warstwy ukrytej, a potem musi opanować umiejętność rekonstrukcji pełnych danych wejściowych z tej skompresowanej informacji. Ponieważ w sieci autoasocjacyjnej nie ma żadnych bezpośrednich połączeń między warstwą wejściową a wyjściową to jedynym sposobem wiernego odtworzenia sygnałów wejściowych na wyjściu jest właśnie znalezienie sposobu kompresji sygnału między warstwą wejściową a ukrytą i sposobu rekonstrukcji sygnału między warstwą ukrytą a warstwą wyjściową. Oznacza to, że sieć autoasocjacyjna w trakcie uczenia zdobywa

umiejętność redukcji wymiaru wejściowych danych. Wystarczy teraz po nauczeniu sieci takiego przekształcenia, odrzucić ostatnią warstwę, a warstwa do tej pory ukryta, stanie się wyjściową i wygeneruje sygnał 3-krotnie skompresowany (w przypadku sieci MLP 9:3:9).

Zwykle zadanie nieliniowej kompresji danych wejściowych (i nieliniowej ich dekompresji na wyjściu) jest zadaniem zbyt złożonym, by mogło być ono wykonane z użyciem sieci o jednej warstwie ukrytej. Z tego powodu sieć autoasocjacyjna jest najczęściej siecią MLP z trzema warstwami ukrytymi, z których środkowa wyznacza reprezentację o zredukowanym wymiarze. Dwie pozostałe warstwy ukryte są potrzebne do przeprowadzenia przez sieć dwóch skomplikowanych transformacji nieliniowych: transformacji wejść do postaci skompresowanej, dostępnej w środkowej warstwie ukrytej, oraz odpowiednio transformacji danych pojawiających się w środkowej warstwie ukrytej do postaci wyjściowej. Sieć autoasocjacyjna posiadająca tylko jedną warstwę ukrytą może realizować wyłącznie liniową redukcję wymiaru i w rzeczywistości uczy się aproksymacji standardowej PCA.

Do realizacji zadania wyznaczenia optymalnego wektora wejściowego przy pomocy sieci autoasocjacyjnej zostały wybrane arbitralnie dwa stopnie kompresji: 56→27 (2,1) oraz 56→19 (2,9) przy wykorzystaniu sieci MLP w postaci: 56-42-**27**-42-56 i 56-40-**19**-40-56. Najlepszymi sieciami dla realizacji preprocessingu z wykorzystaniem autoasocjatorów okazały się kolejno modele MLP: 27:8:1 i 19:3:1. Statystyki regresyjne dla tych sieci przedstawione są w tabeli 11.

	<i>Najlepsza sieć 27:8:1</i>			<i>Średnia z 10 sieci 27:(2-8):1</i>			<i>Najlepsza sieć dla 19:3:1</i>		
	<i>Tr</i>	<i>V</i>	<i>Te</i>	<i>Tr</i>	<i>Ve</i>	<i>T</i>	<i>Tr</i>	<i>V</i>	<i>Te</i>
<i>Odch. Błędu</i>	27.405	29.898	34.217	28.572	30.086	33.449	29.458	31.079	30.884
<i>Śr. bł. bezwz.</i>	20.312	21.767	24.937	21.060	21.840	24.557	21.409	22.774	23.025
<i>RMS</i>	0.0147	0.016	0.0184	0.0179	0.0191	0.0209	0.0197	0.0208	0.0207
<i>SD ratio</i>	0.086	0.104	0.084	0.090	0.105	0.083	0.093	0.108	0.076
I^2	0.00029	0.00036	0.00040	0.00032	0.00037	0.00038	0.00034	0.00039	0.00032
I	1.710%	1.893%	2.004%	1.789%	1.928%	1.954%	1.838%	1.967%	1.802%
I_1^2	0.000%	0.015%	0.764%	0.003%	0.023%	0.652%	0.000%	0.026%	0.031%
I_2^2	0.334%	0.014	6.178%	0.239%	0.011%	5.988%	0.222%	0.000%	0.545%
I_3^2	99.665%	99.971%	93.058%	99.758%	99.967%	93.360%	99.778%	99.974%	99.424%
U	0.995			1.018			1.018		

Tabela 11. Statystyki regresyjne dla autoasocjacyjnej sieci MLP.

Nietrudno zauważyć, na podstawie tabeli 11, że zastosowanie preprocessingu w postaci nieliniowej kompresji wektora wejściowego przyniosło nieoczekiwane najgorsze jak dotąd rezultaty, porównywalne z metodą Głównych Składowych. Można zatem przypuszczać, że dane zawierają pewien statystycznie istotny szum, którego usunięcie (zarówno przy pomocy liniowej jak i nieliniowej kompresji) powoduje pogorszenie możliwości prognostycznych modelu.

3.3.7. Wyznaczenie optymalnego wektora wejściowego przy pomocy selekcji krokowej wstecznej i postępującej.

Algorytmy krokowe, wstecznej i postępującej selekcji cech, dodają lub usuwają po jednej zmiennej w każdym kroku. Algorytm postępujący znajduje na początku taką zmienną, która ma najkorzystniejszy wpływ na poprawę jakości modelu. Proces ten jest kontynuowany aż do osiągnięcia braku poprawy jakości modelu, przy dołączaniu kolejnych zmiennych. Algorytm wstecznej selekcji pracuje w przeciwnym kierunku. Proces rozpoczyna się z uwzględnieniem wszystkich zmiennych. W kolejnych krokach usuwa się po jednej zmiennej, zaczynając od tej spośród nich, której brak najmniej psuje jakość predykcji [51].

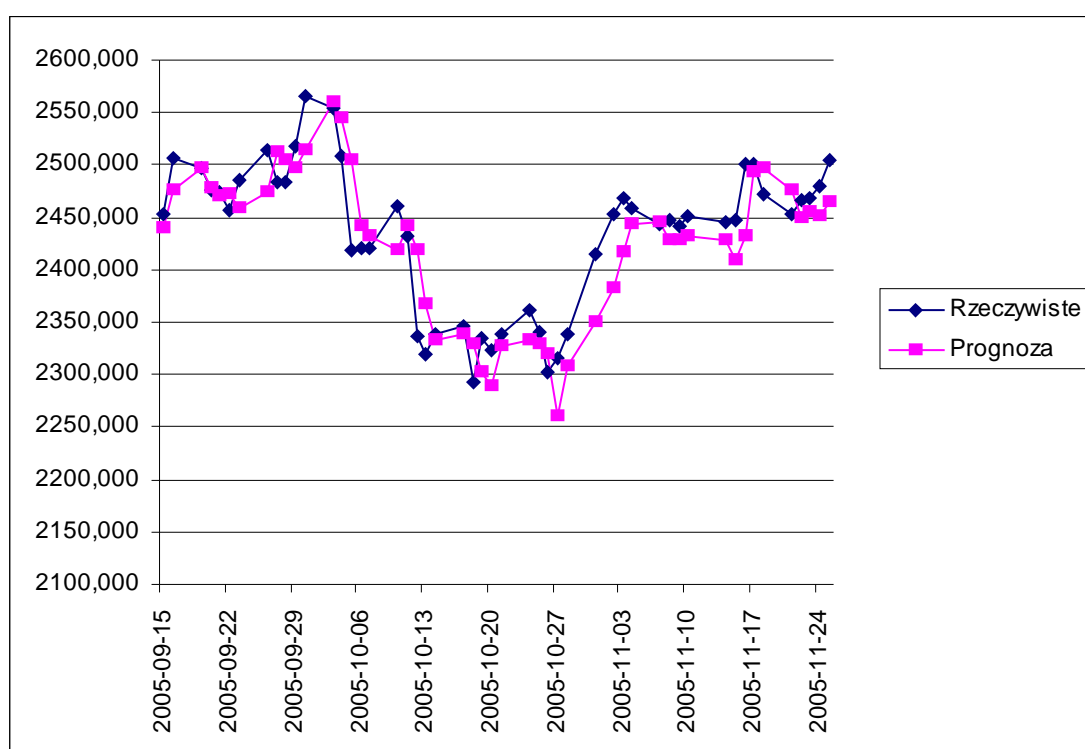
Przy zastosowaniu algorytmów zarówno wstecznej jak i postępującej selekcji krokowej, przestrzeń wejść została ograniczona do 26 zmiennych. Pozostały wszystkie zmienne dotyczące indeksu WIG20 oraz WIGu, średnie WIG20 i średnie wykładnicze, średni obrót, indeksy SP500, a także zmienne dotyczące indeksu BUX. Pozostałe zmienne zostały usunięte.

Statystyki regresyjne dla tak otrzymanego modelu przedstawia poniższa tabela.

	<i>Najlepsza sieć dla 26:6:1</i>			<i>Średnia 10 sieci 26:4-7:1</i>		
	<i>Tr</i>	<i>V</i>	<i>Te</i>	<i>Tr</i>	<i>Ve</i>	<i>T</i>
<i>Odch. Błędu</i>	27.018	27.480	28.356	27.859	27.359	28.109
<i>Śr. bł. bezwz.</i>	19.813	20.104	21.051	20.423	19.904	21.097
<i>RMS</i>	0.0181	0.0184	0.0199	0.0187	0.0183	0.0190
<i>SD ratio</i>	0.085	0.096	0.070	0.087	0.095	0.070
I^2	0.00028	0.00030	0.00027	0.00030	0.00030	0.00028
<i>I</i>	1.686%	1.740%	1.654%	1.739%	1.733%	1.659%
I_1^2	0.000%	0.094%	0.000%	0.003%	0.138%	0.235%
I_2^2	0.198%	0.062%	0.852%	0.217%	0.038%	2.440%
I_3^2	99.802%	99.844%	99.148%	99.780%	99.824%	97.326%
<i>U</i>	0.927			0.946		

Tabela 12. Statystyki regresyjne dla sieci uzyskanej przy pomocy Selekcji krokowej.

Statystyki regresyjne zamieszczone w tabeli 12 są nieznacznie gorsze od tych uzyskanych przy pomocy Algorytmu Genetycznego jednak prawie identyczne jak przy modelu bez preprocessingu danych wejściowych. Jest jednak jedna niezmiernie ważna różnica. Mianowicie przy pomocy algorytmu selekcji krokowej zostało usuniętych aż 30 spośród 56 pierwotnych zmiennych wejściowych. Pozostały zatem te zmienne, które mają najistotniejszy wpływ na zmienną wyjściową i jednocześnie są prawdopodobnie najmniej zaszumione. Można zauważyć również, że średnie parametry regresyjne z wielu utworzonych sieci neuronowych nie odbiegają w sposób znaczący od parametrów regresyjnych dla najlepszej sieci neuronowej.



Rys. 9. Rzeczywiste i prognozowane wartości indeksu WIG20 w okresie 15-09-05 do 25-11-05.

3.4. Podsumowanie

Podsumowując można stwierdzić, że rozważając zagadnienie optymalnego doboru zmiennych wejściowych, nie każda z przedstawionych powyżej metod preprocessingu danych korzystnie wpływa na możliwości prognostyczne omawianego modelu. Okazuje się, że arbitralnie dobrany wektor danych wejściowych prawie w całości jest powiązany ze zmienną zależną w postaci wartości indeksu WIG20. Można jednak przypuszczać, że dane cechuje duża redundancja, gdyż przy zastosowaniu preprocessingu w postaci krokowej selekcji, algorytm usunął aż 30 zmiennych i co ciekawe nie wpłynęło to na pogorszenie statystyk

regresyjnych w porównaniu z modelem bez preprocessingu danych wejściowych. Lepsze rezultaty przyniosło zastosowanie preprocessingu w postaci Analizy Wrażliwości (usunięcie 11 zmiennych) i Automatycznego Projektanta (usunięcie 26 zmiennych). Niestety w obu tych przypadkach następowały duże różnice w doborze wektora wejściowego przy każdym ponownym uruchomieniu wymienionych algorytmów. Pewnym najlepszym „złotym środkiem” może wydawać się Algorytm Genetyczny. Co prawda usunął on tylko dwie zmienne (ATR14 - średnią rzeczywistego zasięgu i dynamikę zmiany logarytmicznych dwudniowych stóp zwrotu), jednak tak uzyskane statystyki regresyjne dla najlepszej sieci były porównywalne z tymi uzyskanymi przy pomocy Automatycznego Projektanta i Analizy Wrażliwości, a średnie statystyki regresyjne dla 10 sieci były najlepsze spośród wszystkich średnich i bardzo zbliżone do tych najlepszych.

Zdecydowanie najgorzej w rozważanym modelu prognostycznym wypadł preprocessing danych wejściowych przy pomocy metody Głównych Składowych i nieliniowej kompresji danych w postaci autoasocjacyjnej sieci MLP.

4. Prognozowanie wartości indeksu WIG20

Tematem rozważań niniejszego rozdziału będzie pokazanie możliwości prognostycznych sieci neuronowych w oparciu o analizę szeregów czasowych i próbę ich zastosowania do budowy strategii inwestycyjnej, a także sprawdzenie możliwości prognostycznych modelu MLP (z rozdziału 3) w prognozowaniu wartości indeksu WIG20.

4.1. Prognozowanie wartości indeksu WIG20 na podstawie jego wartości z poprzednich okresów – typowa neuronowa analiza szeregów czasowych.

Celem analizy szeregów czasowych jest ustalenie prognozy przyszłych wartości pewnej zmiennej, o wartościach zmieniających się w czasie. Najczęściej prognozę oblicza się w oparciu o wcześniejsze wartości tej samej zmiennej, której wartość ma być prognozowana. Niemniej jednak często sięga się również po inne zmienne niezależne, jeśli mogą mieć one wpływ na badaną zmienną (tak jak zostało to przedstawione w poprzednim rozdziale, z zaproponowanymi 56 zmiennymi wejściowymi).

Najczęściej prognozuje się następną wartość szeregu czasowego na podstawie określonej liczby wartości poprzedzających lub innych danych wejściowych. W takim przypadku chodzi o wyznaczenie wartości oddalonej w czasie o jeden krok do przodu. Jednak nic nie stoi na przeszkodzie by w podobny sposób wyznaczać prognozy długookresowe. Z pomocą sieci neuronowych można bowiem wyznaczać wartości oddalone o dowolną liczbę kroków, jednak praktyka pokazuje, że prognozy długookresowe są mniej pewne, a przez to mniej użyteczne niż te dotyczące najbliższego okresu czasu (patrz tabela 1).

Prognozy ponadto można wydłużać w czasie poprzez wieloetapową prognozę szeregu czasowego, która polega na tym, że po odgadnięciu pierwszej nieznannej wartości szeregu czasowego można kontynuować prognozowanie kolejnych wartości poprzez wprowadzenie oszacowanej zmiennej na wejście sieci. Takie wieloetapowe prognozowanie można kontynuować, jednak należy mieć świadomość, że wraz ze zwiększaniem odległości między wartością bieżącą i prognozowaną zmniejsza się niezawodność prognozy. Kiedy jednak potrzebna jest nam prognoza na określoną liczbę okresów do przodu, wówczas najwłaściwszym rozwiązaniem będzie stworzenie sieci dostosowanej stricte do realizacji takiej prognozy, a nie stosowanie prognozy iterowanej. Jednak w rozważanym przypadku, kiedy mowa jest o wykorzystaniu sieci neuronowych do generowania strategii

inwestycyjnych, tworzenie prognoz na więcej niż jeden okres do przodu raczej pozbawione jest sensu, głównie z powodu dużych wahań i zmienności kursów akcji.

Stosowanie modeli neuronowych w analizie szeregów czasowych danych ekonomicznych ma już swoją ponad piętnastoletnią historię. Można zatem wskazać na wiele prac poświęconych tej tematyce, na przykład [2], [3], [67], [75], a zwłaszcza związanych z giełdą [18].

4.1.1. Dobór zmiennych wejściowych.

W modelach neuronowych przeznaczonych do analizy szeregów czasowych jako dane wejściowe wykorzystywane są przeszłe wartości szeregu. Tworząc sieć należy zatem określić ile przypadków (obserwacji wcześniejszych wartości rozważanej zmiennej) musi zostać wprowadzonych na wejście sieci, aby w oparciu o nie sieć mogła wyznaczyć wartość wyjściową, czyli kolejną prognozowaną wartość tej zmiennej. Wielkość ta określana jest czasami jako *rząd procesu*. Przy wyborze wielkości rzędu procesu pojawia się jednak pewien dylemat. Z jednej strony im więcej opóźnionych danych wprowadzimy na wejście sieci, tym budowany model może być „mądrzejszy”, a przez to bardziej skuteczny. Jednak jest też i druga strona medalu. Zwiększanie liczby wejść zwiększa rozmiar sieci, a to powoduje wzrost liczby jej parametrów, co może być przyczyną zmniejszenia zdolności sieci do generalizacji. W większości przypadków proces doboru wartości wejściowych modelu może być wykonany wyłącznie na drodze empirycznej. Bezpośredni dobór wejść możliwy jest tylko w nielicznych przypadkach, kiedy znana jest nam struktura badanego szeregu czasowego, lecz wówczas zazwyczaj nie ma potrzeby budowy modelu neuronowego.

W niniejszym rozdziale przedstawione zostaną dwa konkurencyjne podejścia. W pierwszym prognozować będziemy wartość indeksu na zamknięciu tylko w oparciu o przeszłe wartości szeregu. Natomiast drugie podejście to prognozowanie wartości indeksu WIG20 z wykorzystaniem optymalnego wektora zmiennych wejściowych (54 dane wejściowe) uzyskanego przy pomocy algorytmu genetycznego.

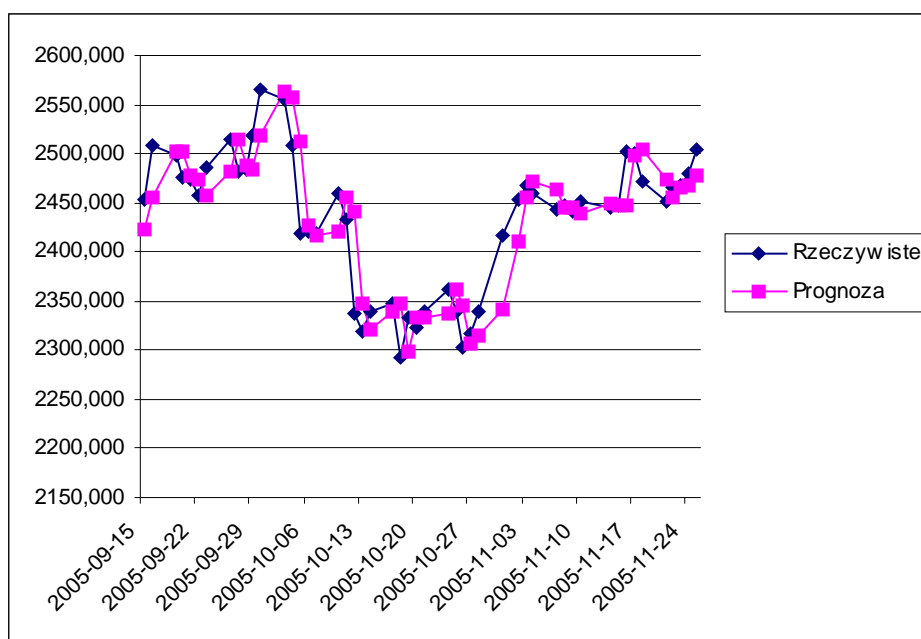
4.1.2. Model prognostyczny.

Istnienie modelu prognostycznego indeksów giełdowych, który na podstawie tylko danych opóźnionych daje w wyniku dobre prognozy jest mało prawdopodobny, gdyż przeczyłoby to teorii, która mówi, iż „rynek dyskontuje wszystko”, czyli każda informacja, każde dane makroekonomiczne, wszystkie czynniki, które mają wpływ na kurs, są już w nim

uwzględnione. Twierdzenie to dotyczy zarówno całych rynków, których zachowanie reprezentują indeksy, jak i pojedynczych akcji o dostatecznie wysokich obrotach.

	Najlepsza MLP s4 1:4-5-1:1			Najlepsza Liniowa s4 1:4-1:1			Najlepsza RBF s4 1:4-96-1:1		
	Tr	V	Te	Tr	Ve	T	Tr	V	Te
Odch. Błędu	28.250	28.099	34.383	30.405	28.209	28.185	27.492	30.633	32.982
Śr. bł. Bezwz.	20.734	20.083	23.759	21.496	20.480	21.197	20.242	21.757	23.709
RMS	0.0181	0.018	0.022	0.0196	0.0182	0.0181	0.0002	0.0003	0.0003
SD ratio	0.086	0.086	0.096	0.091	0.086	0.081	0.081	0.098	0.095
I^2	0.00031	0.00029	0.00045	0.00035	0.00031	0.00029	0.00029	0.00037	0.00040
I	1.749%	1.713%	2.122%	1.877%	1.762%	1.716%	1.694%	1.915%	1.998%
I_1^2	0.020%	0.019%	0.181%	0.000%	0.524%	0.000%	0.001%	0.386%	0.778%
I_2^2	0.015%	0.025%	0.124%	0.209%	0.366%	3.212%	0.216%	0.041%	0.004%
I_3^2	99.965%	99.957%	99.695%	99.791%	99.110%	96.788%	99.783%	99.573%	99.218%
U	0.9999			1.0003			0.994		

Tabela 13. Statystyki regresyjne dla neuronowego modelu prognostycznego z danymi opóźnionymi.



Rys. 10. Rzeczywiste i prognozowane wartości indeksu WIG20 w okresie 15-09-05 do 25-11-05.

	Rzeczywiste	Prognozowane
2005-10-06	2421,120	2426,666
2005-10-07	2419,700	2416,873
2005-10-10	2460,080	2420,818
2005-10-11	2432,180	2455,035
2005-10-12	2336,380	2440,452
2005-10-13	2318,770	2346,578
2005-10-14	2338,460	2320,433
2005-10-17	2346,820	2339,668
2005-10-18	2293,230	2347,042
2005-10-19	2334,170	2298,881
2005-10-20	2323,230	2333,401
2005-10-21	2339,410	2333,958
2005-10-24	2360,770	2337,371
2005-10-25	2340,240	2362,753
2005-10-26	2302,530	2345,568
2005-10-27	2316,550	2306,589
2005-10-28	2339,280	2315,109
2005-10-31	2415,790	2341,025
2005-11-02	2452,650	2410,931
2005-11-03	2468,590	2455,619
2005-11-04	2459,420	2470,849

Tabela 14. Przykładowe zestawienie rzeczywistych i prognozowanych wartości indeksu WIG20.

Tabela 13 przedstawia statystyki regresyjne dla prognostycznych modeli neuronowych, w których danymi wejściowymi są wartości indeksu WIG20 z poprzednich okresów. Jak się tego można było spodziewać, otrzymane wyniki są dużo gorsze od wyników uzyskanych przy pomocy modeli, które jako wektor wejściowy oprócz danych opóźnionych zawierały również inne zmienne.

Najlepszymi sieciami okazały się sieci: MLP s4 1:4-5-1:1, Liniowa s4 1:4-1:1 i RBF s4 1:4-96-1:1, dla których rząd procesu (ilość opóźnionych danych wejściowych) wynosi 4. Jednak otrzymane przy pomocy tych sieci prognozy wartości indeksu WIG20 o jeden okres do przodu są praktycznie zupełnie nieprzydatne. Okazuje się bowiem, iż równie dobrą prognozą dla okresu $t+1$ może być wartość indeksu z okresu t . Świadczy o tym statystyka *U Theila*, której wartość dla powyższych modeli jest równa jedności. Można zatem stwierdzić, iż próba budowy efektywnego systemu decyzyjnego na podstawie powyższego modelu staje się bezsensowna, a dodanie większej ilości danych wejściowych, które mogą mieć wpływ na zmienną prognozowaną staje się koniecznością, co dobrze obrazuje rozdział 3 niniejszej pracy, który pokazuje jak zmienia się efektywność tworzonych prognoz wartości indeksu WIG20 przy pomocy modeli neuronowych, w zależności od przyjętego wektora zmiennych wejściowych. Na szczególną uwagę zasługują wyniki uzyskane przy pomocy

modelu z 54-elementowym wektorem zmiennych wejściowych (zob. rozdział 3.2.5), który różni się od modelu pierwotnego tym, iż poprzez usunięcie przy pomocy algorytmu genetycznego 2 zmiennych niezależnych (średniej rzeczywistego zasięgu i dynamiki zmiany logarytmicznych dwudniowych stóp zwrotu) uzyskano zauważalną poprawę statystyk regresyjnych omawianego modelu prognostycznego.

4.2. Prosta strategia decyzyjna w oparciu o prognozę wartości indeksu WIG20 z wykorzystaniem wielu zmiennych wejściowych.

Nasuwa się pytanie, czy na podstawie modelu prognostycznego z rozdziału 3.2.5 można skonstruować model decyzyjny wspomagający podejmowanie trafnych decyzji inwestycyjnych. Punktem wyjścia niech będą estymowane przy pomocy tego modelu wartości indeksu WIG20 z końca sesji. Najprostsza strategia jaka się nasuwa jest następująca: mając prognozowaną wartość indeksu w czasie $t+1$ porównujemy ją z wartością rzeczywistą indeksu w chwili t . Jeśli prognozowana wartość indeksu na jeden okres naprzód jest większa od wartości indeksu w czasie t to powinniśmy zakupić walory wchodzące w skład WIG20. Analogicznie w przeciwnym przypadku powinniśmy takie walory sprzedać. W przypadku takiego modelu decyzyjnego zysk z inwestycji byłby imponujący nawet po odjęciu prowizji maklerskiej. Niestety tutaj pojawia się problem. Tworząc prognozę na okres $t+1$ korzystamy z danych wejściowych, których wartość znana jest nam dużo później niż czas zamknięcia Giełdy Papierów Wartościowych w Warszawie (np. amerykański indeks SP500 i NASDAQ). Nawet pominięcie tych parametrów nie rozwiązuje sprawy, gdyż niemożliwe jest jednoczesne wyestymowanie przyszłej wartości indeksu na podstawie wartości z zamknięcia sesji oraz niemożliwe jest kupienie lub sprzedanie walorów po cenie zamknięcia.

Można jednak nieznacznie zmodyfikować powyższą strategię, tak aby była ona możliwa do realizacji, co przedstawia poniższa tabela 15.

Data	Wartość rzeczywista zamknięcia	Prognoza	Wartość rzeczywista otwarcia	Strategia I	Zysk lub strata	Zakumulowany zysk	Strategia II	Przychód
2000.01.03	1852,9	1806,7	1862,9	K	-100	-100	0	0
2000.01.04	1796,6	1823,2	1818,2	S	216	116	S	18182
2000.01.05	1777,0	1792,2	1739,0	S	-380	-264	0	0
2000.01.06	1832,1	1812,7	1776,2	K	559	295	K	-35524
2000.01.07	1933,2	1858,0	1867,0	K	662	957	0	0
2000.01.10	2000,8	1991,4	1968,8	K	320	1277	0	0
2000.01.11	1974,9	1991,2	1977,4	S	25	1302	S	39548
2000.01.12	1961,3	1958,0	1940,5	S	-208	1094	0	0
2000.01.13	2032,9	1959,3	1979,8	S	-531	563	0	0
2000.01.14	2010,5	2061,2	2041,1	K	-306	257	K	-40822
2000.01.17	2027,0	1986,0	2001,5	S	-255	2	S	40030
2000.01.18	1982,6	2023,5	2032,3	S	497	499	0	0
2000.01.19	1971,9	2008,8	1953,6	K	183	682	K	-39072
2000.01.20	1995,2	2004,9	1985,8	K	94	776	0	0
2000.01.21	2003,0	1996,6	1975,0	K	280	1056	0	0
2000.01.24	1933,1	2005,7	1982,3	K	-492	564	0	0
2000.01.25	1873,7	1899,0	1898,6	S	249	813	S	37972
2000.01.26	1914,1	1913,0	1896,1	K	180	993	K	-18961
2000.01.27	1951,3	1926,6	1910,0	K	413	1406	0	0

Tabela 15. Wyniki dwóch strategii inwestycyjnych opartych na prognozie wartości indeksu WIG20 na zakończeniu sesji, przy założeniu, że istnieje instrument o cechach kontraktu terminowego, ale o wartości całkowicie zgodnej z wartością indeksu.

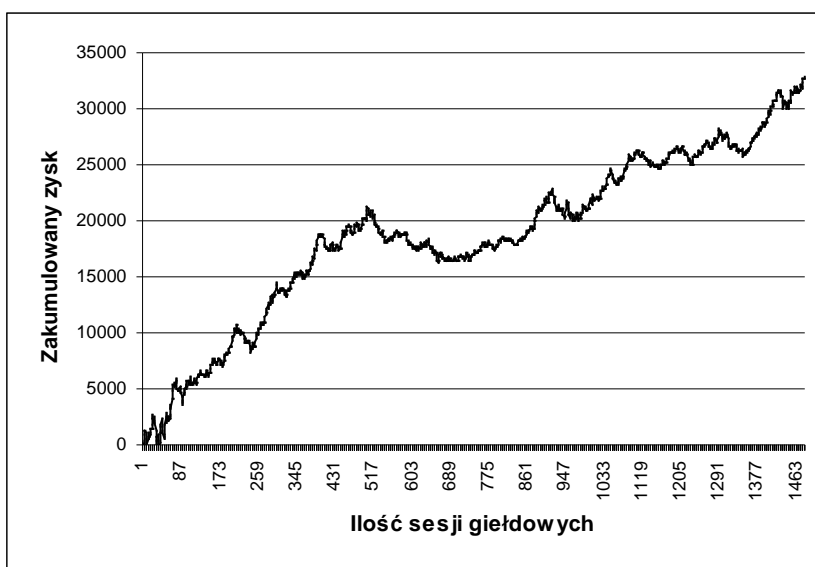
Powyższa tabela 15 przedstawia dwie strategie inwestycyjne oparte na rozważanym neuronowym modelu prognostycznym. Pierwsze trzy kolumny to kolejno: data, rzeczywista i prognozowana wartość indeksu WIG20 na zamknięciu sesji, zaś kolumna czwarta zawiera rzeczywistą wartość indeksu na otwarciu sesji.

Przedstawiona w kolumnie nr 5 strategia inwestycyjna zbudowana jest na następującej zasadzie: Jeżeli prognozowana wartość indeksu dla czasu $t+1$ będzie większa od rzeczywistej wartości indeksu w czasie t to należy dokonać zakupu instrumentu pochodnego na otwarciu sesji w czasie $t+1$ i sprzedać go na końcu sesji po cenie zamknięcia. Analogicznie w przeciwnym przypadku - jeżeli prognozowana wartość indeksu dla czasu $t+1$ będzie mniejsza od rzeczywistej wartości indeksu w czasie t to należy dokonać krótkiej sprzedaży⁵ instrumentu pochodnego na otwarciu sesji w czasie $t+1$ i odkupić go pod koniec sesji po cenie zamknięcia. Używamy tutaj pojęcia instrumentu pochodnego, który z założenia

⁵ Krótka sprzedaż jest to sprzedaż danego waloru pomimo zerowego stanu posiadania. Krótka sprzedaż akcji jest to sprzedaż akcji pożyczonych od maklera, czyli zajęcie pozycji z ujemną ich liczbą. Obecnie na Giełdzie Papierów Wartościowych w Warszawie krótka sprzedaż akcji nie jest możliwa. Możliwa jest jednak krótka sprzedaż kontraktów terminowych na akcje (kontrakty terminowe na akcje największych spółek są notowane na GPW w Warszawie) lub na indeksy. Posiadanie pozycji krótkiej (krótko sprzedanych walorów) jest zyskowne w przypadku spadku ich notowań.

posiadałby cechy kontraktu terminowego⁶ ale o wartości całkowicie zgodnej z wartością instrumentu bazowego, jaką jest wartość indeksu WIG20. Jest to niestety założenie czysto teoretyczne, które sprawia, że wynik z inwestycji w WIG20 nie ma wartości merytorycznej, a jedynie czysto pogładową, pozwalającą na porównanie wszystkich opracowanych w niniejszej pracy modeli.

Zakumulowany zysk przy tak sformułowanej strategii decyzyjnej z 1482 sesji giełdowych z okresu od 03.01.2000r. do 25.11.2005r. wynosi 32773 PLN, co przedstawia poniższy rysunek 11. Należy jednak podkreślić, że podany powyżej zakumulowany zysk nie uwzględnia prowizji maklerskiej od zawartych transakcji kupna sprzedaży. Nawet przy założeniu najniższej możliwej prowizji od kupna/sprzedaży kontraktu terminowego w wysokości 15 PLN czyli 30 PLN w trakcie 1 sesji (najniższa prowizja maklerska na rynku akcji wynosi 0,2% od wartości transakcji) ponieśliśmyby zakumulowane koszty w wysokości: 44460 PLN, co uczyniłoby zaproponowaną strategię zupełnie nieopłacalną.



Rys. 11. Zakumulowany zysk uzyskany na podstawie strategii I bez uwzględnienia prowizji maklerskiej od przeprowadzanych transakcji.

Można również zaproponować nieznaczny modyfikację strategii inwestycyjnej poprzez zmniejszenie liczby zawieranych transakcji kupna sprzedaży (co obrazuje strategia II). Polega ona na tym, że utrzymujemy bieżącą pozycję do momentu zmiany strategii I, np. utrzymujemy zakupione po cenie otwarcia walory do momentu, aż w strategii I nastąpi zmiana z K na S, a wówczas krótko sprzedajemy instrument pochodny po cenie otwarcia. W ten sposób nastąpiła redukcja zawieranych transakcji z 1482 do 614, co zredukowało

⁶ Obszerny opis giełdowego kontraktu terminowego (Futures), jego istotę oraz standaryzację przedstawia **dodatek B** do niniejszej rozprawy.

koszty do 18420 PLN. Jednocześnie jednak spadł zakumulowany zysk do 11770 PLN (spadek zakumulowanego zysku wynika z różnicy pomiędzy ceną zamknięcia w czasie t , a ceną na otwarciu sesji w czasie $t+1$), co czyni zaproponowaną strategię równie nieopłacalną jak poprzednia.

4.3. Prognozowanie wartości zamknięcia z przesunięciem w czasie.

Wykorzystanie prognozowanych wartości indeksu WIG20 przy pomocy modelu z podrozdziału 3.2.5 do wspomagania podejmowania decyzji inwestycyjnych nie jest ekonomicznie uzasadnione, co zostało wykazane w poprzednim podrozdziale. Powodem tego jest trudność, jaką napotyka potencjalny inwestor z wykorzystaniem otrzymanej prognozy, a z drugiej strony jej stosunkowo niska skuteczność. Ten drugi negatywny czynnik można w pewnym stopniu osłabić poprzez pewien ciekawy zabieg na danych wejściowych. Do tej pory estymując wartość indeksu na zamknięcie sesji w czasie $t+1$ używaliśmy wszystkich danych wejściowych z czasu t . Okazuje się jednak, że przynajmniej niektóre z danych wejściowych mogą być użyte również z okresu $t+1$. Mianowicie dotyczy to takich zmiennych, których czas powstania jest wcześniejszy lub taki sam co godzina otwarcia GPW w Warszawie (wartość indeksów: DAX, BUX, WIG, WIG20 na otwarciu). To podejście nie jest niestety idealne i pozbawione problemów (o czym napiszemy obszerniej w dalszej części podrozdziału), jednak jakość otrzymanych prognoz jest znacząco lepsza w porównaniu z poprzednim modelem, co obrazuje poniższa tabela 16.

	<i>Najlepsza MLP 54-4-1</i>		
	Tr	V	Te
<i>Odch. Błędu</i>	19.146	20.363	21.349
<i>Śr. bł. Bezwz.</i>	14.788	15.362	16.347
<i>RMS</i>	0.0128	0.0137	0.0143
<i>SD ratio</i>	0.060	0.071	0.053
I^2	0.00014	0.00017	0.00016
I	1.195%	1.295%	1.248%
I_1^2	0.000%	0.887%	0.157%
I_2^2	0.092%	0.036%	3.632%
I_3^2	99.908%	99.077%	96.210%
U	0.6727		

Tabela 16. Statystyki regresyjne dla neuronowego modelu prognostycznego z przesunięciem w czasie.

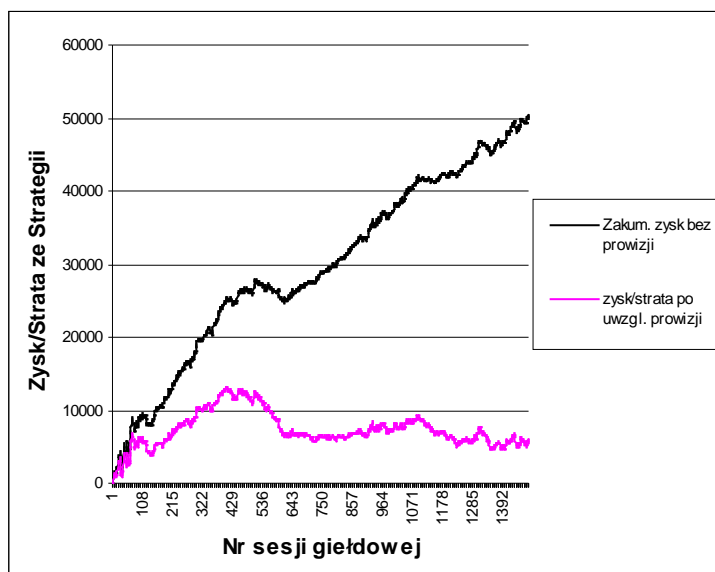
W rozważanym modelu średniokwadratowy błąd prognozy dla $t+1$ wynosi poniżej 1,5%, a odpowiedzi sieci różnią się średnio od rzeczywistych wartości prognozowanych o około 15-16 jednostek (o czym informuje nas średni błąd bezwzględny). Przeciętny względny błąd prognozy w badanym okresie nie przekracza 1,3%, co jest wynikiem imponującym.

Przy rozważanym modelu zostaną zastosowane analogiczne strategie decyzyjne jak w poprzednim podrozdziale. Otrzymane wycinkowe wyniki przedstawia poniższa tabela 17.

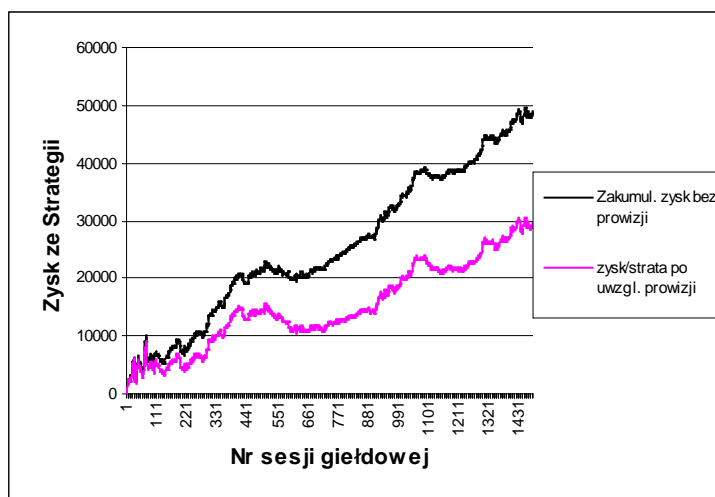
Data	Wartość rzeczywista zamknięcia	Prognoza	Wartość rzeczywista otwarcia	Strategia I	Zysk lub strata	Zakumulowany zysk	Strategia II	Przychód
2000.01.03	1852,9	1884,7	1862,9	K	-10,0	-10,0	-	0
2000.01.04	1796,6	1805,0	1818,2	S	21,6	11,6	S	1818,2
2000.01.05	1777,0	1756,1	1739,0	S	-38,0	-26,4	Utrzymuj	0
2000.01.06	1832,1	1801,2	1776,2	K	55,9	29,5	K	-3552,4
2000.01.07	1933,2	1897,5	1867,0	K	66,2	95,7	Utrzymuj	0
2000.01.10	2000,8	1969,1	1968,8	K	32,0	127,7	Utrzymuj	0
2000.01.11	1974,9	1947,3	1977,4	S	2,5	130,2	S	3954,8
2000.01.12	1961,3	1931,0	1940,5	S	-20,8	109,4	Utrzymuj	0
2000.01.13	2032,9	1989,5	1979,8	K	53,1	162,5	K	-3959,6
2000.01.14	2010,5	2033,6	2041,1	K	-30,6	131,9	Utrzymuj	0
2000.01.17	2027,0	1982,1	2001,5	S	-25,5	106,4	S	4003,0
2000.01.18	1982,6	2004,8	2032,3	S	49,7	156,1	Utrzymuj	0
2000.01.19	1971,9	1964,3	1953,6	S	-18,3	137,8	Utrzymuj	0
2000.01.20	1995,2	2005,8	1985,8	K	9,4	147,2	K	-3971,6
2000.01.21	2003,0	1984,1	1975	S	-28,0	119,2	S	3950,0
2000.01.24	1933,1	1980,1	1982,3	S	49,2	168,4	Utrzymuj	0
2000.01.25	1873,7	1908,9	1898,6	S	24,9	193,3	Utrzymuj	0
2000.01.26	1914,1	1928,2	1896,1	K	18,0	211,3	K	-1896,1
2000.01.27	1951,3	1935,5	1910	K	41,3	252,6	Utrzymuj	0

Tabela 17. Wyniki dwóch strategii inwestycyjnych opartych na prognozie wartości indeksu WIG20 na zakończeniu sesji, przy założeniu, że istnieje instrument o cechach kontraktu terminowego, ale o wartości całkowicie zgodnej z wartością indeksu.

Zakumulowany zysk z rozważanego modelu prognostycznego i strategii decyzyjnej jak z przykładu poprzedniego z 1482 sesji giełdowych z okresu od 03.01.2000r. do 25.11.2005r. wynosi 50067 PLN, a po zastosowaniu strategii nr II, mającej na celu ograniczenie ilości zawieranych transakcji nieznacznie spada do wartości 48246 PLN. Należy jednak podkreślić, że podane wartości zakumulowanego zysku nie uwzględniają kosztów transakcyjnych w postaci prowizji maklerskiej. Zakumulowane zyski z obydwu strategii inwestycyjnych z uwzględnieniem kosztów transakcyjnych przedstawiają poniższe wykresy.



Rys. 12. Zakumulowany zysk uzyskany na podstawie strategii I.



Rys. 13. Zakumulowany zysk uzyskany na podstawie strategii II.

Na podstawie powyższych wykresów okazuje się, że tylko Strategia nr II przynosi rosnący w czasie zysk. Co prawda Strategia nr I również przyniosła zysk jednak nie ma on tendencji do wzrostu w dłuższym okresie czasu, co może być spowodowane zbiegiem okoliczności – ale tę hipotezę trzeba by było dokładniej zbadać (nie zrobiono tego w ramach niniejszej pracy).

Na tym etapie pojawia się jednak dodatkowy problem, co zostało już wcześniej zasygnalizowane. Rozważając powyższy model prognostyczny i oparte na nim dwie strategie decyzyjne nie wzięto pod uwagę opóźnienia czasowego. Mianowicie prognozując wartość indeksu WIG20 na zamknięciu sesji w czasie $t+1$ korzystamy z niektórych danych (wartość indeksów: DAX, BUX, WIG, WIG20 na otwarciu) również z okresu $t+1$. Zakładając nawet, że wszystkie wartości czterech danych wejściowych będą dostępne bez ograniczeń,

a prognoza wartości indeksu zostanie wyestymowana w czasie rzeczywistym, to może nie udać się potencjalnemu inwestorowi dokonać zakupu zakładanego instrumentu pochodnego po cenie otwarcia, w szczególności gdy na giełdzie dojdzie do gwałtowniejszych reakcji innych uczestników. Oznacza to, że w rzeczywistości inwestorowi może nie udać się osiągnąć nawet takich zysków jak zaprezentowane w niniejszym rozdziale, bo rzeczywistym rynkiem rządzą nieco bardziej złożone prawa, niż te które zakładano w opisywanym tu wyidealizowanym modelu.

4.4. Podsumowanie.

Podsumowując niniejszy rozdział można stwierdzić, że w oparciu o neuronowy model prognostyczny można otrzymać dość dobre prognozy przyszłych wartości ekonomicznych, w naszym przypadku - wartości indeksu WIG20 na zamknięciu sesji. Okazuje się jednak, że o ile uzyskanie takiej prognozy jest problemem stosunkowo prostym, o tyle stworzenie na jej podstawie systemu doradczego wspomagającego decyzje potencjalnego inwestora, nie jest już zagadnieniem trywialnym. Pojawia się tu bowiem problem z interpretacją otrzymanych wyników jak również z ich potencjalnym wykorzystaniem. Dlatego zamiast odrębnie opracowywać model, który dostarczy prognozowanych wartości rozważanych parametrów, a potem odrębnie budować system komputerowy, który pomoże decydentowi w podjęciu właściwej decyzji podejmiemy (w kolejnym rozdziale) próbę budowy systemu doradczego, który będzie od razu tworzył rady inwestycyjne i sygnały kupna-sprzedaży.

5. Neuronowe modele decyzyjne

Realizacja efektywnej strategii inwestycyjnej na giełdowym rynku akcji, walut, czy kontraktów terminowych, chociaż często oparta jest na wspieranej doświadczeniem intuicji inwestora (co prowadzi z reguły do dość przypadkowych efektów), powinna zasadniczo bazować na odpowiednio skonstruowanym **modelu decyzyjnym**. Wobec braku jawnych i możliwych do zidentyfikowania relacji, wiążących dostępne na rynku akcji informacje z przyszłą dynamiką rynku, tworzenie tego typu modeli poprzez dobór *a priori* odpowiednich równań lub reguł logicznych na ogół nie jest celowe. Budowa efektywnych modeli decyzyjnych zazwyczaj oparta jest na analizie i przetwarzaniu **historycznych** danych, opisujących zachowanie się kursów akcji w przeszłości.

W sytuacji, gdy w analizowanym systemie brak jest możliwości identyfikacji jawnych reguł opisujących dynamikę systemu, a dodatkowo występują w nim znaczne zakłócenia, natomiast dostępna jest duża ilość zaobserwowanych danych, sprawdzonym narzędziem służącym do modelowania tego typu systemów są sieci neuronowe. Opisane wyżej przesłanki występują wyraźnie w systemach finansowych takich jak rynki akcji, wskazując na celowość zastosowania metodyki wspomaganie podejmowania decyzji opartej na sztucznych sieciach neuronowych właśnie w zagadnieniach modelowania i wspomaganie decyzji finansowych. Efektywność wykorzystania tych metod na rynkach finansowych została również potwierdzona przez liczne badania, a także publikacje dokumentujące praktyczne rezultaty zastosowań [5][6][57][58][59][67].

W zdecydowanej większości wspomnianych wyżej zastosowań, jako narzędzie do tworzenia modeli decyzyjnych stosowane są sieci uczone w sposób nadzorowany (np. perceptron lub RBF), przy czym zmienną wyjściową (której realizacje w ciągu uczenia stanowią wzorcowe wartości wyjściowe sieci) jest z reguły pewna wielkość (np. stopa zwrotu, obserwowany trend, itp.) charakteryzująca dynamikę zmian kursu walorów w odpowiedzi na zadane wartości zmiennych wejściowych.

5.1. Porównanie sieci neuronowych typu MLP oraz RBF.

W niniejszym rozdziale do tworzenia modeli decyzyjnych zostaną zastosowane dwa rodzaje sieci neuronowych. Omawiane wcześniej i wielokrotnie wykorzystywane wielowarstwowe sieci perceptronowe (MLP), a także sieci o radialnych funkcjach bazowych (RBF). Sieci neuronowe MLP odgrywają z matematycznego punktu widzenia rolę narzędzi

do aproksymacji stochastycznej funkcji wielu zmiennych, odwzorowując zbiór zmiennych wejściowych $x \in R^N$ w zbiór zmiennych wyjściowych $y \in R^M$ [29][35].

Inny komplementarny sposób rozumienia odwzorowania zbioru wejściowego w wyjściowy, to odwzorowanie przez dopasowanie funkcji aproksymującej wielu zmiennych do wartości żądanych, czyli rozciągnięcie nad zbiorem uczącym hiperpowierzchni (w wybranych przypadkach - hiperpłaszczyzny wielowymiarowej) najlepiej dopasowującej się do wartości żądanych. Neurony ukryte, ze względu na rolę jaką pełnią, stanowią zbiór funkcji będących bazą dla wektorów wejściowych x .

Jest to więc odmienna koncepcja w podejściu do sztucznych sieci neuronowych, polegająca na zastosowaniu nieliniowego przekształcenia przestrzeni danych wejściowych przez wprowadzenie warstwy neuronów ukrytych, realizujących funkcję zmieniającą się radialnie wokół wybranego centrum. Funkcje takie są nazywane radialnymi funkcjami bazowymi, a rola neuronu ukrytego, sprowadza się do odwzorowania jedynie niewielkiego fragmentu przestrzeni danych wokół jednego punktu (lub grupy punktów stanowiących klaster).

Sieci typu radialnego stanowią naturalne uzupełnienie sieci sigmoidalnych. Dzieje się tak dlatego, iż neuron sigmoidalny reprezentował w przestrzeni wielowymiarowej hiperpłaszczyznę⁷ separującą tę przestrzeń na dwie klasy, natomiast neuron radialny reprezentuje hipersferę (lub hiperelipsoide), dokonującą podziału kołowego wokół punktu centralnego. W przypadku wystąpienia symetrii kołowej danych zastosowanie sieci radialnych stwarza możliwość znaczącego zmniejszenia liczby neuronów potrzebnych do realizacji zadania klasyfikacyjnego. Ponadto, w sieciach radialnych nie istnieje potrzeba stosowania wielu warstw ukrytych, a to za sprawą odmiennej funkcji, jaką w tej sieci pełnią neurony ukryte. Udowodniono bowiem [16], że przyjęcie dostatecznie dużej liczby neuronów ukrytych realizujących funkcję radialną zapewnia rozwiązanie problemu klasyfikacyjnego przy użyciu dwu warstw sieci: warstwy ukrytej i warstwy wyjściowej o jednym neuronie liniowym. Wynika to stąd, że jeśli neuronów w warstwie ukrytej jest dostatecznie dużo, to można do każdego istotnego szczegółu modelowanej funkcji przywiązać odpowiedni neuron radialny, co gwarantuje, że uzyskane rozwiązanie będzie odwzorowywało zadaną funkcję z całkowicie satysfakcjonującą wiernością.

Typowa neuronowa sieć radialna jest zatem strukturą zawierającą warstwę wejściową, na którą są podawane sygnały wejściowe x , warstwę ukrytą o neuronach radialnych i warstwę

⁷ W ogólnym przypadku powierzchnia ta ma bardziej złożony kształt i hiperpłaszczyzna może jedynie stanowić jej aproksymację. Taka bardziej złożona powierzchnia dzieląca przestrzeń sygnałów wejściowych przy stosowaniu sieci klasy MLP nazywana bywa w literaturze „urwiskiem sigmoidalnym”.

wyjściową złożoną zwykle z jednego neuronu, którego zadaniem jest sumowanie sygnałów pochodzących od neuronów ukrytych i wytwarzanie potrzebnego sygnału wyjściowego. Tworzone są też sieci radialne o wielu wyjściach, ale są one zdecydowanie mniej popularne.

Sieci neuronowe o radialnych funkcjach bazowych znalazły zastosowanie zarówno w rozwiązywaniu problemów klasyfikacyjnych, w zadaniach aproksymacji funkcji wielu zmiennych, jak i w zagadnieniach predykcji, czyli w tym obszarze, gdzie sieci jednokierunkowe, a w szczególności sieci sigmoidalne mają od lat ugruntowaną pozycję. Sieci radialne w porównaniu do sieci wielowarstwowych o sigmoidalnych funkcjach aktywacji wyróżniają się pewnymi szczególnymi właściwościami, które umożliwiają lepsze odwzorowanie cech charakterystycznych modelowanego procesu.

Sieć sigmoidalna reprezentuje aproksymację globalną zadanej funkcji (jest to efekt sigmoidalnej funkcji aktywacji, która rozciąga się od pewnego punktu w przestrzeni aż do nieskończoności), podczas gdy sieć radialna, bazująca na funkcjach mających niezerową wartość jedynie w określonej przestrzeni wokół centrów, realizują aproksymację typu lokalnego. Ma to bezpośrednie przełożenie na zdolności uogólniania sieci, w których zazwyczaj gorzej wypada sieć RBF, zwłaszcza na granicach obszaru danych uczących albo poza tymi granicami (gdy potrzebna jest ekstrapolacja wyuczonych wyników).

Sieci radialne natomiast, bardzo dobrze radzą sobie z problemem minimów lokalnych. Funkcje radialne typu gaussowskiego, najczęściej używane w praktyce, są z natury funkcjami lokalnymi o wartości niezerowej jedynie wokół określonego centrum. Umożliwia to łatwe powiązanie parametrów funkcji bazowych z fizycznym rozmieszczeniem danych uczących w przestrzeni wielowymiarowej. Sprawia to, iż stosunkowo łatwo można uzyskać dobre wartości startowe, a to z wielokrotnia prawdopodobieństwo sukcesu w procesie uczenia i osiągnięcie minimum globalnego funkcji błędu.

Przestrzenie decyzyjne tworzone w sieciach radialnych są stosunkowo proste i w sposób naturalny ukształtowane. Przy rozwiązywaniu zadań klasyfikacyjnych dla danych testujących, różniących się od danych uczących sieć radialna dostarcza nie tylko informacji, do jakiej klasy należy dany wzorzec testujący, ale również wskazuje na ewentualną możliwość utworzenia oddzielnej klasy. Jest to problem szczególnie istotny w tych zagadnieniach, gdzie nie ma pewności, że w uczeniu zastosowano dostateczną liczbę wzorców uczących, a rozkład wzorców testujących jest inny niż wzorców uczących. Przykładem może być zastosowanie sieci neuronowych w medycynie, do wykrywania komórek rakowych w organizmach żywych. Aktualna baza ucząca może nie odpowiadać

danym testującym, w których pojawi się nowy rodzaj komórki rakowej. Sieć radialna może wskazać na utworzenie nowego typu zwyrodnienia.

Powszechna w literaturze [48][73] jest opinia, że sieci radialne lepiej niż sieci MLP oparte na neuronach o sigmoidalnych charakterystykach, nadają się do takich zadań klasyfikacyjnych jak: wykrywanie uszkodzeń w różnego rodzaju systemach, rozpoznawanie wzorców, itp. Ponadto zastosowanie sieci RBF w predykcji skomplikowanych szeregów czasowych również umożliwia uzyskanie porównywalnych lub nawet lepszych rezultatów niż przy zastosowaniu sieci sigmoidalnych.

Ważną zaletą sieci radialnych jest znacznie uproszczony algorytm uczenia, co wynika z bliższego usytuowania punktu startowego do rozwiązania optymalnego, w porównaniu z sieciami wielowarstwowymi. Dodatkowo możliwe jest odseparowanie etapu doboru parametrów funkcji bazowych od doboru wartości wag sieci, co może również uprościć i przyspieszyć proces uczenia. Jeśli dodatkowo uwzględnimy procedurę kształtowania optymalnej pod względem zdolności uogólniania struktury sieci, to zysk na czasie będzie znacznie większy. Dzieje się tak dlatego, że w sieciach wielowarstwowym jest to zagadnienie bardzo czasochłonne, wymagające zwykle wielokrotnego uczenia lub douczania. Natomiast w sieciach RBF proces optymalnego kształtowania struktury sieci jest stałym fragmentem uczenia, nie wymagającym żadnego dodatkowego wysiłku.

Wspólny pozostaje jedynie problem doboru liczby neuronów ukrytych, decydujący w bardzo dużym stopniu o dokładności odwzorowania i zdolnościach uogólniania sieci. W przypadku sieci radialnej jest on jednak o wiele prostszy niż w sieciach sigmoidalnych, ze względu na lokalny charakter aproksymacji reprezentowany przez poszczególne funkcje bazowe.

5.2. Krótkoterminowe neuronowe strategie decyzyjne.

W niniejszym podrozdziale zostaną zaprezentowane dwa modele. Pierwszy model służy do generowania decyzji spekulacyjnej jaką należy podjąć na początku danej sesji. Prognozujemy zatem czy nastąpi wzrost/spadek wartości indeksu od otwarcia do zamknięcia sesji w następnym dniu.

Możliwe jest wystąpienie 3 stanów:

- pozycja długa – kupno na otwarciu sesji,
- pozycja krótka – sprzedaż na otwarciu sesji

- poza rynkiem – wstrzymanie się z kupnem i sprzedażą na danej sesji. Ta opcja występuje wówczas, gdy zmiana wartości indeksu jest mniejsza od założonej (w tabeli jako parametr „x”).

Parametr „x” określa jaka duża musi być punktowa zmiana wartości indeksu, aby zająć odpowiednią pozycję na otwarciu giełdy innej niż „poza rynkiem”. Określa zatem, jakie musi być minimalne wahanie indeksu na danej sesji, aby pojawił się sygnał kupna lub sprzedaży.

Te trzy możliwe do wystąpienia stany tworzą wektor danych wyjściowych (dane wzorcowe).

W omawianym modelu mamy do czynienia z klasyfikacją (na podstawie danych wejściowych, takich jak w poprzednich rozdziałach, ustalamy jeden z trzech stanów na kolejną sesję). Przy problemach klasyfikacji dobre rezultaty przynosi zastosowanie sieci RBF, a jej zaletą (o czym była mowa wcześniej) jest powtarzalność wyników, szybkość uczenia i uzyskiwanie minimów globalnych. Do budowy modelu została użyta sieć neuronowa RBF o 236 neuronach radialnych. Wektor danych wejściowych składał się z 1183 danych treningowych i 300 walidacyjnych dobranych w sposób losowy.

Uzyskane rezultaty obrazują poniższe tabele.

X	100 sesji	1 rok	2 lata	3 lata	5 lat	6 lat
15	380	1160	1890	4040	5060	26340
12	110	580	20	990	2290	25150
10	180	-40	-630	-1410	-2940	18880
9	-1720	-1250	-4500	-4710	-11170	9070
8	-620	-1070	-5440	-9370	-17800	-3150
7	-780	-1320	-6050	-10980	-18130	-2480
6	-540	-650	-4840	-8800	-22250	-5870
5	-2350	-5350	-11360	-13810	-23570	-7250
4	-2750	-5620	-12520	-18520	-31030	-8590
3	-1390	-3650	-6170	-10330	-21050	-3280
2	-2020	-3560	-8450	-14510	-25570	-7150
1	290	-2110	-7700	-12270	-19620	-1790
0	-2300	-5200	-7650	-7730	-19160	2530

Tabela 18. Hipotetyczny zysk z inwestycji w indeks WIG20 od otwarcia do zamknięcia sesji w zależności od parametru „x”.

W tabeli 18 zobrazowany jest hipotetyczny zysk z inwestycji w indeks WIG20, po uwzględnieniu prowizji w wysokości 15 pln (30 pln w trakcie 1 sesji) za zajęcie pozycji zgodne z prognozą uzyskaną przy pomocy sieci neuronowej. Optymalny zysk osiągany jest przy parametrze „x” z przedziału od 12 do 15. Jest to niestety tylko czysto teoretyczne

podejście, przy założeniu, że istnieje instrument o cechach kontraktu terminowego, ale o wartości całkowicie zgodnej z wartością indeksu. Jest to zatem założenie czysto teoretyczne, co sprawia, że wynik z inwestycji w WIG20 nie ma wartości merytorycznej, a jedynie czysto poglądową.

Porównanie tak zdefiniowanego najprostszego modelu decyzyjnego z analogicznym modelem z rozdziału 4.2, opartego na prognozie wartości indeksu na zamknięciu sesji, pokazuje nam zdecydowaną różnicę w zyskowności (tu osiągamy zysk w wysokości 26 tys. przy stracie na poprzednim modelu rzędu kilkunastu tysięcy). Problemem jaki pojawia się w obecnym modelu jest nierównomierne rozłożenie zysków w czasie.

Drugi model jest analogicznie jak w rozdziale 4.2 z nieznaczną modyfikacją pierwszej strategii decyzyjnej. Celem jest redukcja ilości zawieranych transakcji, poprzez utrzymywanie bieżącej pozycji (kupionych lub krótko sprzedanych instrumentów na otwarciu sesji), aż do kolejnej zmiany odpowiedzi sieci neuronowej. Dzięki temu zabiegowi potencjalny hipotetyczny zysk z inwestycji wzrósł w perspektywie 6 letniej nawet do 67390 PLN przy parametrze x równym 0, co jest wynikiem znacząco lepszym niż w modelach z rozdziału 4. Nieznacznie zmniejszyła się również dysproporcja w rozkładzie zysków w czasie.

X	100 sesji	1 rok	2 lata	3 lata	5 lat	6 lat
15	4880	5750	7370	11680	14680	58550
12	4650	5560	6090	9490	15940	64380
10	5850	7290	9450	12070	19310	65870
9	1600	3620	4150	8500	13640	57730
8	4940	7670	8690	8560	15340	53270
7	4650	7200	8400	11180	19320	62930
6	3550	6990	8880	12120	16320	56620
5	3440	3390	3190	9070	16780	54550
4	3300	3580	4290	5270	10520	58680
3	3630	5940	10750	16640	24050	66670
2	2690	6360	9860	11730	17660	60410
1	5670	8920	11400	13580	21750	62230
0	3290	6950	11820	17940	24720	67390

Tabela 19. Hipotetyczny zysk z inwestycji w indeks WIG20 przy zastosowaniu Strategii nr II w zależności od parametru „ x ”.

Dotychczas używaliśmy modeli opartych na hipotetycznym instrumencie pochodnym, który z założenia posiadałby cechy kontraktu terminowego. Istnieje jednak alternatywa dla takiego sposobu przeprowadzania transakcji, a mianowicie kupno lub sprzedaż kontraktów

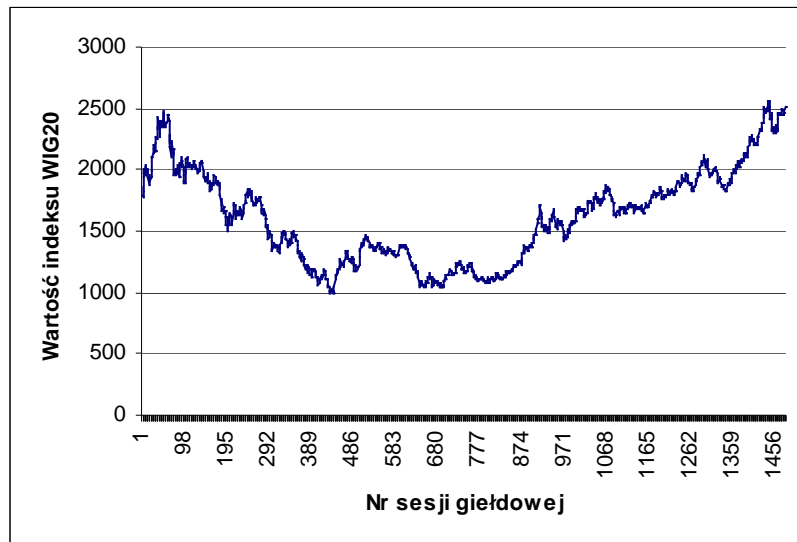
terminowych na indeks WIG20. Inwestowanie, a w zasadzie spekulacja przy pomocy kontraktów terminowych daje inwestorowi możliwość uzyskania dużo wyższej stopy zwrotu niż w przypadku inwestowania w akcje, gdyż inwestor spekulujący na kontraktach korzysta z dźwigni finansowej. W tym momencie mówimy już nie o hipotetycznych, ale o rzeczywistych zyskach jakie można było osiągnąć przy pomocy neuronowej strategii decyzyjnej na danych historycznych.

x	100 sesji	1 rok	2 lata	3 lata	5 lat	6 lat
15	5530	6300	7870	12450	15660	52350
12	5210	5860	5910	9070	14560	57400
10	6310	7760	10090	13280	22340	61180
9	2900	4600	4590	9090	14020	48150
8	4960	7040	8400	8080	15360	45930
7	5110	7650	8620	11730	19220	55880
6	4360	7240	8540	12140	14390	45870
5	3740	3120	2590	8690	15890	48080
4	3820	3980	5850	7670	12950	53090
3	5000	6960	11770	18780	27530	61590
2	2350	5750	7800	10250	16380	52250
1	5970	8820	11790	14570	24280	57430
0	3650	7190	12140	19100	27250	62300

Tabela 20. Realny zysk z inwestycji w kontrakty terminowe na indeks WIG20 przy zastosowaniu Strategii nr II w zależności od parametru „x”.

Dociekliwy czytelnik w tym momencie zada dwa pytania: dlaczego w powyższych modelach występuje tak wartościowo duża różnica pomiędzy horyzontem inwestycji 5 i 6 letniej oraz jaki można by osiągnąć zysk w przypadku inwestycji pasywnej (kupić lub krótko sprzedać i utrzymywać taką pozycję przez cały rozważany okres). Odpowiedzią na to pytanie niech będzie poniższy wykres. Widać na jego podstawie, że w przeciągu rozważanych pierwszych 100 sesji nastąpił bardzo gwałtowny ruch wartości indeksu z 1777 punktów w czasie t_1 do 2482 w czasie t_{50} i potem ponowny spadek do wartości 1897 na 100 sesji. Trafne przewidzenie tego ruchu indeksu poprzez sieć neuronową było najbardziej prawdopodobną przyczyną tak dużej różnicy między zyskami osiągniętymi w perspektywie 5 i 6 lat.

Natomiast pasywny zysk potencjalnego inwestora zarówno w perspektywie 5 jak i 6 lat wyniósłby około 7000 PLN. Można zatem stwierdzić, iż neuronowa strategia decyzyjna w postaci zaprezentowanej w niniejszym rozdziale przynosi znacząco lepsze rezultaty od możliwego do osiągnięcia w analogicznym czasie zysku pasywnego.



Rys. 14. Wartość indeksu WIG20 w perspektywie 6 letniej.

Obserwując otrzymane wyniki, na podstawie Strategii nr II można zauważyć, że najlepsze wyniki inwestycyjne można było osiągnąć (zarówno w przypadku inwestycji hipotetycznej jak i realnej przy pomocy kontraktów terminowych) stosując model z parametrem „x” równym 0. Oznacza to, iż w efekcie stosujemy strategię, w której występują tylko dwa stany: pozycja długa i krótka, a wyeliminowany natomiast zostaje całkowicie stan – poza rynkiem, gdyż zmiana wartości indeksu na kolejnych sesjach jest różna od zera.

Mając tak otrzymany optymalny model działający na danych historycznych, możemy dokonać ekstrapolacji strategii na dane najnowsze. Do tej pory bowiem wszystkie wyliczone zyski opierały się na danych historycznych podzielonych na dwie grupy: walidacyjną i testową. Metodologia przeprowadzanej ekstrapolacji będzie analogiczna do rzeczywistej inwestycji na giełdzie przy pomocy omawianego modelu. Mamy więc do dyspozycji 100 „przyszłych” przypadków, które nie były dotychczas wykorzystane przy uczeniu sieci. Tworzenie prognozy składa się zatem ze 100 kroków. Nie mamy tu jednak do czynienia z blokiem 100 przypadków testowych. Nasza predykcja polega na tym, iż prognozujemy o 1 okres naprzód, poczym przypadek testowy staje się uczącym i tworzymy prognozę na kolejny dzień. Zaznaczyć również należy, że dla każdego przypadku wykonujemy kilka prób z losowym rozłożeniem przypadków na uczące i walidacyjne, a otrzymana decyzja jest średnią z kilku prób.

Otrzymane wyniki przedstawia poniższa tabela.

	Inwestycja w WIG20		Inwestycja w kontrakty	
	Bez uwzględnienia prowizji	Po odjęciu prowizji	Bez uwzględnienia prowizji	Po odjęciu prowizji
Strategia aktywna	5396	4376	5040	4020
Strategia pasywna	5828	5798	5650	5620

Tabela 21. Zestawienie zysków osiągniętych przy pomocy neuronowej strategii decyzyjnej dla danych najnowszych z okresu 29-11-2005 do 20-04-2006

Tabela 21 zawiera porównanie zysków osiągniętych przy pomocy omawianej strategii inwestycyjnej z możliwymi do osiągnięcia zyskami ze strategii pasywnej. Z jednej strony osiągnięty zysk jest imponujący – ponad 4000 PLN w 5 miesięcy (100 sesji), z drugiej jednak strony czytelnik zapewne zauważy, iż zysk osiągnięty przy pomocy inwestycji pasywnej przewyższa o 1600 PLN zysk z modelu neuronowego. Tak się bowiem złożyło, iż w rozważanym okresie mamy do czynienia z trendem wzrostowym, a nie wszystkie sygnały uzyskane przy pomocy sieci neuronowej były trafne (62% odpowiedzi sieci było trafnych, co jest zadowalającym rezultatem). Ponadto należy również pamiętać, iż decydując się na inwestycje nie wiemy czy najbliższe 100 sesji będzie wzrostowych czy spadkowych, a co udało się tu dowieść, omawiany model trafnie to przewidział.

5.3. Neuronowe strategie decyzyjne oparte o prognozowany trend.

Na zakończenie, w niniejszym podrozdziale zaprezentowane zostaną dwie neuronowe strategie inwestycyjne, których celem będzie generowanie sygnałów kupna i sprzedaży dla potencjalnego inwestora w oparciu o krótkoterminowy trend. Tym, co odróżnia ten model od poprzednich, jest właśnie prognozowanie trendu, a nie zmiany wartości indeksu na 1 sesji. Możliwe są dwa podejścia w wygenerowaniu wektora danych wyjściowych (wzorcowych) opisującego istniejący trend. Pierwsze to wyliczenie przy pomocy określonej funkcji – charakterystyki występującego na giełdzie trendu (co obrazuje pierwszy z omawianych modeli), drugie podejście to empiryczne, częściowo subiektywne określenie przebiegu trendu w oparciu o wiedzę i intuicję inwestora, a także jego preferencje dotyczące akceptowalnego przez niego ryzyka, na podstawie historycznego wykresu wartości indeksu.

Możliwe jest wystąpienie 2 stanów, stanowiących wzorzec danych wyjściowych:

- pozycja długa (sygnał kupna) – kupno na otwarciu sesji,
- pozycja krótka (sygnał sprzedaży) – sprzedaż na otwarciu sesji.

W tym modelu dana pozycja jest utrzymywana również po zamknięciu sesji (pozwala to zmniejszyć wartość prowizji), aż do wygenerowania przez sieć neuronową sygnału pozycji przeciwstawnego do aktualnie zajmowanego. Wzorzec danych wyjściowych został utworzony w następujący sposób: Najpierw zostało obliczone nachylenie trendu⁸ wartości otwarcia indeksu WIG20, a następnie na tej podstawie utworzono następującą regułę decyzyjną:

- **Jeżeli nachylenie trendu > 0 (a akcje są krótko sprzedane) to dokonaj kupna akcji (K),**
- **Jeżeli nachylenie trendu ≤ 0 (a akcje są kupione) to dokonaj krótkiej sprzedaży akcji (S).**

Do uczenia sieci zostało użytych 2230 przypadków, z czego 1784 to dane uczące, a 446 dane walidacyjne wyznaczane w sposób losowy. Dane pochodzą z okresu od 02-01-1997 do 28-11-2005, a przy budowie modelu zostały przeanalizowane dwa rodzaje sieci neuronowych: Perceptronowa i Sieć o radialnych funkcjach bazowych. Wyniki zawarte są w poniższych tabelach.

	RBF					MLP				
	100 sesji	1 Rok	3 Lata	5 Lat	6 Lat	100 sesji	1 Rok	3 Lata	5 Lat	6 Lat
3 Okr.	2900	7520	17120	30230	29650	4530	7310	12720	19650	26610
4 Okr.	2270	4990	18520	28750	33840	3610	3270	7040	19160	30000
5 Okr.	3410	7740	23930	34790	41340	3450	5000	12410	22430	23780
6 Okr.	3850	7160	23940	38560	41230	3850	7380	24820	39840	40370
7 Okr.	3260	7110	21280	29800	33920	2920	6640	21190	30710	32540

Tabela 22. Hipotetyczny zysk z inwestycji w indeks WIG20 z uwzględnieniem prowizji od zawieranych transakcji.

	RBF					MLP				
	100 sesji	1 Rok	3 Lata	5 Lat	6 Lat	100 sesji	1 Rok	3 Lata	5 Lat	6 Lat
3 Okr.	3520	7740	17350	30150	29800	4470	7040	12180	19890	24960
4 Okr.	2920	5250	19260	31100	35080	3460	3620	6030	20660	28880
5 Okr.	4320	7990	24650	35220	40200	3750	5210	11310	21130	21400
6 Okr.	3850	7380	24820	39840	40370	3960	6220	10150	15730	16320
7 Okr.	2920	6640	21190	30710	32540	80	1660	11240	16140	16930

Tabela 23. Realny zysk z inwestycji w kontrakty terminowe na indeks WIG20 z uwzględnieniem prowizji od zawieranych transakcji.

Zyski przedstawione w powyższych tabelach zostały wyznaczone jako średnie zyski z kilku prób. Widać zatem, że zyski uzyskane przy pomocy sieci RBF są znacząco lepsze

⁸ Nachylenie trendu było obliczone jako współczynnik kierunkowy prostej regresji dla kilku ostatnich notowań kursu otwarcia dla indeksu WIG20.

zarówno dla hipotetycznej inwestycji w indeks WIG20 jak i realnej inwestycji w kontrakty terminowe od zysków uzyskanych przy pomocy neuronowych modeli decyzyjnych opartych o sieci typu perceptronowego (MLP). Najlepszymi modelami okazały się dwa modele, dla których wzorcem danych wyjściowych był współczynnik kierunkowy prostej regresji dla 5 i 6 (5 Okr. i 6 Okr.) ostatnich notowań kursu otwarcia dla indeksu WIG20. Okazuje się również, że przedstawiony powyżej model decyzyjny, który generuje sygnały kupna/sprzedaży na podstawie analizowanego krótkookresowego trendu daje bardziej zrównoważone w czasie wyniki od modelu zaprezentowanego w podrozdziale 5.1. Dla potencjalnego inwestora ważniejsze jest bowiem uzyskiwanie zysków zrównoważonych i względnie stałych w czasie. To kryterium bardzo dobrze spełnia omawiany powyżej model, dla którego zyski z biegiem czasu wznoszą w sposób liniowy i wynoszą około 7000PLN rocznie.

Przykładową klasyfikację przypadków uczących i walidacyjnych przedstawia tabela 23. Jak widać wszystkie przypadki znalazły się w grupach poprawnie lub błędnie zaklasyfikowanych. Decyduje o tym tzw. Classification Confidence Level (poziom ufności). Służy on do translacji odpowiedzi neuronu z postaci numerycznej do odpowiedniej klasy. Przykładowo dla standardowo ustalonego przez program Statistica Neural Networks poziomu ufności na poziomie 0,05 przypadki na wyjściu pomiędzy 0,95 a 0,05 klasyfikowane są jako nieznane, a pozostałe klasyfikowane do odpowiedniej klasy. W przypadku tak standardowo dobranego poziomu ufności, przy problemie inwestycji na giełdzie sieć neuronowa nie jest w stanie dokonać klasyfikacji danych do odpowiednich klas, a prawie wszystkie odpowiedzi mają status „nieznany”, czyli niemożliwy do zaklasyfikowania do odpowiedniej klasy. W naszym przypadku Classification Confidence Level został ustalony na poziomie 0,5, tak aby wszystkie przypadki zostały przyporządkowane do odpowiednich klas.

	Dane uczące		Dane walidacyjne	
	d	k	d	k
Total	928	869	247	202
Correct	693	592	184	104
Wrong	235	277	63	98
Unknown	0	0	0	0
d	693	277	184	98
k	235	592	63	104

Tabela 24. Klasyfikacja przypadków dla poziomu ufności 0,5.

Niestety w rozważanym przypadku pojawia się pewien problem. Mianowicie przy próbie ekstrapolacji strategii decyzyjnej na dane najnowsze (tak jak to robiliśmy w poprzednim podrozdziale) uzyskane wyniki są wręcz żenujące i tylko nieznacznie większe

od zera. Można by się zastanawiać, co jest powodem tak słabej zdolności prognostycznej rozważanego modelu. Widać to częściowo w wynikach przedstawionych w tabelach 22 i 23, gdzie w perspektywie 6 lat uzyskane zyski są przeszło o 20 tys. mniejsze od zysków osiągniętych w podrozdziale 5.1. Niewykluczone, iż trend obliczony jako współczynnik kierunkowy prostej regresji zbyt sztywno reaguje na sytuację na rynku, oraz zawiera w sobie niewielką „wiedzę” prognostyczną.

Tych mankamentów pozbawiony jest drugi i zarazem ostatni model prognostyczny. W tym przypadku wzorec danych wyjściowych został utworzony poprzez subiektywną analizę wykresów obrazujących wartości indeksu WIG20.

Do uczenia sieci zostało użyte 1483 przypadki, z czego 1187 to dane uczące, a 296 to dane walidacyjne wyznaczone w sposób losowy. Dane pochodzą z okresu od 03-01-2000 do 28-11-2005.

	Inwestycja w WIG20		Inwestycja w kontrakty	
	Bez uwzględnienia prowizji	Po odjęciu prowizji	Bez uwzględnienia prowizji	Po odjęciu prowizji
100 sesji	5640	5500	5690	5540
1 rok	10150	9490	10590	9930
2 lata	15090	13200	14410	12520
3 lata	20950	18220	20050	17330
5 lat	29290	24370	29900	24980
6 lat	69530	62820	62600	55890

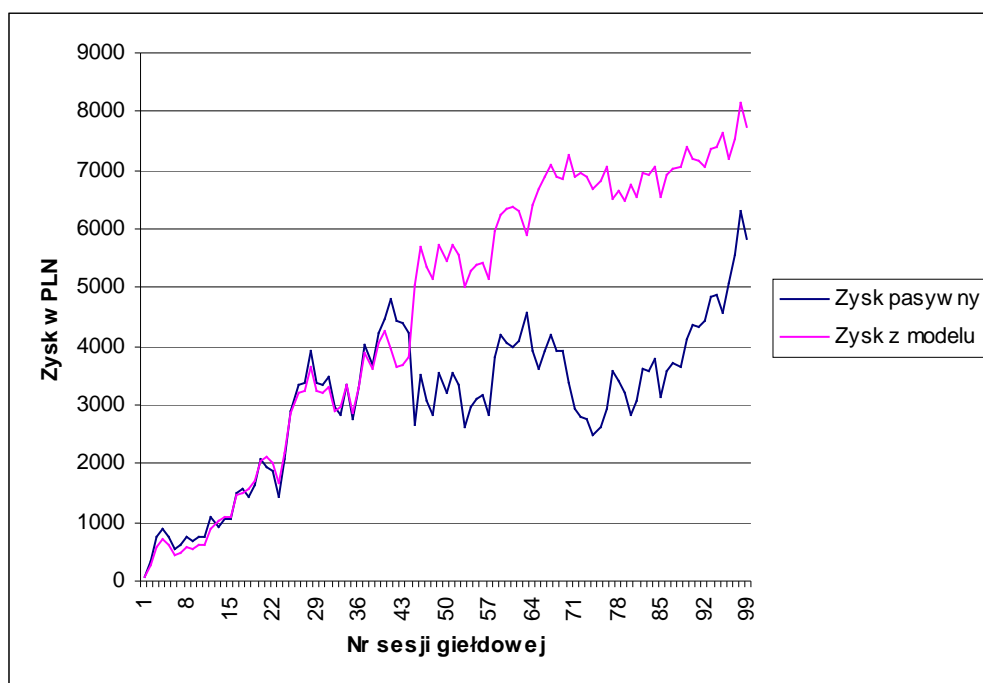
Tabela 25. Zestawienie hipotetycznego oraz realnego zysku z inwestycji wspomagananej neuronowym modelem decyzyjnym w oparciu o krótkoterminowy trend.

Na podstawie powyższej tabeli 25, zawierającej zestawienie zysków z inwestycji wspomagananej neuronowym modelem decyzyjnym w oparciu o krótkoterminowy trend wyznaczony w sposób subiektywny można stwierdzić, że prezentowana strategia przynosi porównywalne zyski jak model z rozdziału 5.1.

Czytelnik powinien w tym momencie zadać pytanie: Jakie zyski można osiągnąć stosując powyższą strategię decyzyjną dla danych najnowszych, oraz jak wygląda porównanie tych zysków z zyskiem osiągniętym w przypadku inwestycji pasywnej (kupieniu na początku i sprzedaniu na końcu rozważanego okresu czasu). Takie zestawienie realnego zysku w okresie od 29-11-2005 do 20-04-2006 w spekulacji na kontraktach terminowych w porównaniu z możliwym do osiągnięcia w analogicznym czasie zyskiem pasywnym przedstawia poniższa tabela oraz wykres. Należy również podkreślić, że metodologia przeprowadzonej symulacji jest analogiczna jak przy strategii z rozdziału 5.1.

	Inwestycja w WIG20		Inwestycja w kontrakty	
	Bez uwzględnienia prowizji	Po odjęciu prowizji	Bez uwzględnienia prowizji	Po odjęciu prowizji
Strategia aktywna	8682	7752	10170	9240
Strategia pasywna	5828	5798	5650	5620

Tabela 26. Zestawienie zysków osiągniętych przy pomocy neuronowej strategii decyzyjnej dla danych najnowszych z okresu 29-11-2005 do 20-04-2006



Rys. 15. Wartość osiągniętego zysku w okresie testowym od 29-11-2005 do 20-04-2006 przy zastosowaniu dwóch strategii pasywnej oraz aktywnej opartej na modelu decyzyjnym.

Jak widać w okresie od 29-11-2005 do 20-04-2006 czyli w przeciągu 100 sesji możliwy do osiągnięcia zysk pasywny wyniósł ponad 5000 PLN, podczas gdy w tym samym czasie realny zysk uzyskany przy pomocy neuronowego modelu decyzyjnego przekroczył wartość 9000 PLN w przypadku spekulacji przy pomocy kontraktów terminowych, co jest wynikiem imponującym. Tak duży zysk osiągnięty w tak krótkim czasie może trochę martwić, gdyż jest on równy średniemu zyskowi osiąganemu na danych historycznych w perspektywie średnio 1 roku. Mamy zatem do czynienia z bardzo dobrym okresem

inwestycyjnym, gdzie dużym wahaniom indeksu przy wzrostowym trendzie towarzyszy duża skuteczność sieci neuronowej w przewidywaniu kierunku zmian. Należy jednak liczyć się z tym, iż jeśli nastąpi gwałtowne odwrócenie trendu model neuronowy może wygenerować krótkoterminową stratę, która uśredni otrzymane wyniki do poziomu wyników uzyskanych na danych historycznych.

5.4. Podsumowanie.

Przedstawione w niniejszym rozdziale modele decyzyjne wspomagające podejmowanie decyzji przez potencjalnego inwestora, a oparte w szczególności na krótkoterminowym trendzie przyniosły imponujący zysk. Co prawda nie we wszystkich przedstawionych modelach uzyskane dochody były wyższe od zysku pasywnego, jednak w istotny sposób różniły się od wyników osiągniętych w rozdziale 4, gdzie mieliśmy do czynienia z prognozą cen uzyskaną przy pomocy sieci neuronowych.

6. Wykorzystanie samoorganizujących sieci neuronowych Kohonena do budowy modeli decyzyjnych.

W niniejszym rozdziale zaprezentowana zostanie metoda generowania strategii transakcyjnych, wykorzystująca sieć neuronową Kohonena (SOM) jako narzędzie wstępnego grupowania wektorów wartości wejściowych, opisujących stan Giełdy Papierów Wartościowych. Jest to podejście jakościowo odmienne od większości technik neuronowych w problemach modelowania rynków finansowych, gdyż wykorzystuje sieć uczoną w sposób nienadzorowany.

6.1. Sieć Kohonena – krótka charakterystyka.

Idea takiego systemu samoorganizującego pojawiła się w latach 50-tych, a terminu tego użyli Farley i Clarck. Sam termin był definiowany bądź jako system, który zmienia swą podstawową strukturę na podstawie informacji docierającej do niego z otoczenia, bądź jako szersze pojęcie obejmujące zarówno złożony obiekt jak i jego otoczenie. Idea samoorganizacji jest ważna, gdyż istnieje dość powszechne przekonanie, iż wszelkie procesy życiowe zawierają w sobie etap samoorganizacji.

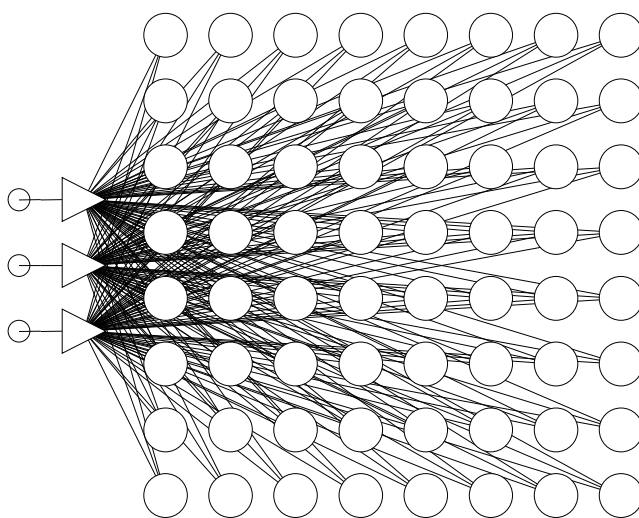
Istotnym elementem, którego istnienie warunkuje możliwość pojawienia się procesu samoorganizacji, jest asymetria istniejących oddziaływań, co oznacza, że jeśli pewne zdarzenie porządkujące ma miejsce, to warunkiem pojawienia się samoorganizacji jest, by oddziaływanie rozbijające porządek w systemie, było słabsze od porządkującego.

Na początek jednak słów parę na temat samych sieci neuronowych typu Kohonena. Sieć Kohonena została nazwana przez jej twórcę [45] samoorganizującym odwzorowaniem, czy też samoorganizującą się mapą cech (Self-Organizing Map – SOM) i jest niewątpliwie najbardziej popularnym typem sieci określanej mianem samoorganizującej.

Sieć Kohonena jest inspirowana przez pewne znane własności mózgu. Kora mózgowa jest w rzeczywistości dużym płaskim płatem komórek nerwowych. Powierzchnia tego płata wynosi około 0,5 metra kwadratowego. Ma ona znaną, pofałdowaną strukturę tylko po to, aby dostosować się do kształtu czaszki. Działanie tego płata związane jest ze znanymi

własnościami topologicznymi - na przykład obszar odpowiadający dłoni znajduje się w nim obok obszaru przedramienia, a ten z kolei sąsiaduje z obszarem ramienia itd. Zatem powierzchnia kory mózgowej jest swoistą mapą topologiczną, odpowiadającą całej złożonej konstrukcji ciała człowieka.

Sieć Kohonena jest jednym z najbardziej znanych typów sieci neuronowej uczonej w trybie bez nauczyciela. Jest siecią o bardzo prostej strukturze (Rys. 16), gdyż posiada tylko dwie warstwy, a przepływ informacji w tej sieci jest ściśle jednokierunkowy. Mimo prostej budowy i nieskomplikowanych metod określających sposób jej funkcjonowania, możliwości aplikacyjne tego typu modelu są olbrzymie.



Rys. 16. Schemat przykładowej sieci Kohonena. Trójkąty symbolizują neurony warstwy wejściowej, a kółka – neurony warstwy wyjściowej.

Sieć Kohonena składa się z dwóch warstw: warstwy wejściowej i warstwy wyjściowej. Neurony wchodzące w skład pierwszej warstwy sieci Kohonena praktycznie nie dokonują przetwarzania danych - posiadają jedynie pojedyncze wejście i pojedyncze wyjście, a jedynym ich celem jest rozesłanie do wszystkich neuronów drugiej warstwy wszystkich wartości wprowadzonych na wejścia sieci (wykorzystuje się tu znowu regułę połączeń typu „każdy z każdym”). Funkcjonalność sieci Kohonena zapewniają więc wyłącznie neurony warstwy wyjściowej zwanej również jako warstwa tworząca mapę topologiczną, ponieważ takie jest jej najczęstsze zastosowanie. Neurony w tej warstwie są rozłożone tak, jakby były rozmieszczone w przestrzeni według jakiegoś ustalonego wzoru - zwykle wyobrażamy je sobie jako węzły dwuwymiarowej siatki, tak jak na rysunku 16.

Sieć Kohonena jest uczona w trybie bez nauczyciela, co oznacza, że wykorzystywany zbiór uczący obejmuje wyłącznie wartości zmiennych wejściowych i nie zawiera żadnych wiadomości dotyczących wartości oczekiwanych na wyjściach sieci. Uczenie ma charakter iteracyjny - zbiór uczący jest wielokrotnie prezentowany sieci. Uczenie rozpoczyna się od losowych, zwykle skupionych wokół zera, wartości współczynników wagowych, przy czym losowe generowanie wartości współczynników wagowych przeprowadza się po ustaleniu struktury sieci, co powoduje, że początkowe wartości wag neuronów sąsiadujących ze sobą nie pozostają w żadnym związku.

W trakcie uczenia sieci prezentowane są kolejne wzorce uczące. Po wprowadzeniu pierwszego z nich wartości wszystkich zmiennych wejściowych wchodzących w skład tego wzorca są przekazywane do wszystkich neuronów warstwy drugiej. Neurony wchodzące w jej skład obliczają swoje sygnały wyjściowe. Każdy neuron drugiej warstwy określa swój sygnał wyjściowy niezależnie od innych neuronów tej warstwy, przy czym wartość sygnału wyjściowego określonego neuronu jest równa odległości Euklidesa pomiędzy wprowadzonym wektorem wartości wejściowych, a wektorem wag rozważanego neuronu. Po przeprowadzeniu tych obliczeń nadrzędny algorytm uczący porównuje wartości wyznaczone na wyjściach poszczególnych neuronów i wybiera ten neuron, dla którego wyznaczona wartość jest najmniejsza (czyli ten, którego wektor wag znajduje się w najmniejszej odległości od wektora wejściowego). Wyznaczony w ten sposób neuron zostaje nazwany neuronem zwycięskim, gdyż on właśnie wygrał współzawodnictwo pomiędzy neuronami warstwy wyjściowej. Jak z tego wynika rywalizacja pomiędzy neuronami sieci Kohonena polega na wyborze takiego neuronu, którego wektor wag jest w największym stopniu zbliżony do wprowadzonego wektora wartości wejściowych.

Po zidentyfikowaniu neuronu zwycięskiego modyfikowane są jego wagi (co jest zasadniczym etapem każdego procesu uczenia). Modyfikacja wag przeprowadzana jest tak, aby neuron zwycięski jeszcze bardziej przybliżył się do wprowadzonego aktualnie wektora wejściowego.

Ostatecznie algorytm Kohonena prowadzi do dopasowania siatki utworzonej przez warstwę wyjściową sieci do obiektów uczących. Oznacza to, że:

- Poszczególne neurony dopasowują się do wzorców uczących czyli prezentacji każdego obiektu towarzyszy uaktywnienie jednego neuronu, posiadającego wektor wag najbardziej zbliżony do wartości zmiennych charakteryzujących obiekt. Często się zdarza, że ten sam neuron jest zwycięzcą dla wielu różnych wektorów wejściowych. Oznacza to, że charakteryzowane przez nie obiekty są podobne, a wektor wag odpowiadającego im

neuronu reprezentuje całą tę grupę. Kolejne pokazy w trakcie uczenia i kolejne korekty prowadzą do tego, że wagi tego neuronu stają się uśrednionymi wzorcami cech charakteryzujących wszystkie obiekty. Sieć Kohonena dzieli więc obiekty zbioru uczącego na klasy, które reprezentowane są przez poszczególne neurony wyjściowe.

W sposób bezpośredni badacz nie może zadać liczby klas, na które podzielone zostaną obiekty zawarte w eksplorowanych danych. Może to jednak zrobić w sposób pośredni - określając strukturę warstwy wyjściowej sieci. Im więcej neuronów będzie wchodzić w jej skład, tym większe będzie rozdrobienie uzyskanych grup. Należy jednak pamiętać, że nie zawsze liczba grup będzie równa liczbie neuronów, gdyż w następstwie samouczenia sieci nie dostajemy nigdy tak dobrych wyników grupowania, jak w przypadku zastosowania uczenia z nauczycielem. Dlatego w sieci Kohonena często pewna liczba neuronów pozostaje niewykorzystana - nie stają się one neuronami zwycięskimi dla żadnego przypadku uczącego.

- Sieć Kohonena zachowuje informację o zależnościach pomiędzy zidentyfikowanymi grupami obiektów. Zależności te głównie odwołują się do pojęcia bliskości (w sensie metryki Euklidesa) odpowiednich wektorów danych, a tę bliskość z kolei możemy kojarzyć z pojęciem podobieństwa opisywanych przez te dane obiektów. Oznacza to, że wraz ze wzrostem podobieństwa pomiędzy wyznaczonymi grupami obiektów, przybliżają się do siebie również reprezentujące je neurony. W efekcie sąsiadujące ze sobą neurony odpowiadają podobnym do siebie grupom danych, zaś neurony oddalone od siebie odpowiadają zróżnicowanym grupom.

Utworzona w trakcie samouczenia sieć Kohonena może posłużyć do zbadania prezentowanego w trakcie uczenia zbioru danych i do ujawnienia takich jego cech, o których istnieniu twórca sieci mógł wcale nie wiedzieć. Nauczona sieć stanowi również pewien model przechowujący wiedzę o strukturze zbiorowości. Może on być wykorzystany dla klasyfikacji i porządkowania także innych danych niż te, które prezentowane były w czasie uczenia. Nauczona sieć Kohonena ma bowiem tę właściwość, że po wprowadzeniu na jej wejścia dowolnych danych (byle tylko tego samego typu, jak wartości podawane na wejścia sieci podczas uczenia), zawsze uaktywni się jakiś neuron wskazujący na klasę, do której najprawdopodobniej powinien zostać przypisany nowy obiekt. Sieć Kohonena może być użyta do klasyfikacji danych również wówczas, gdy klasy wyjściowe są z góry znane. Zaletą takiego z kolei postępowania jest zdolność sieci do przedstawienia (między innymi w formie graficznej) stopnia podobieństwa pomiędzy klasami. W istocie sieci SOM dość często używa

się w zagadnieniu identyfikacji skupień wzorców (formacji) generowanych przez finansowe szeregi czasowe i możliwych do zaobserwowania na ich wykresach, oraz do identyfikacji typowych wzorców (zob. np. [19][50]).

Drugim możliwym zastosowaniem omawianego tu narzędzia jest wykrywanie nowości. Sieć Kohonena może zostać nauczona między innymi tego, by była przystosowana do rozpoznawania skupień w danych uczących i do reagowania na nie. Jeśli na wejściu sieci pojawią się nowe dane, niepodobne do poprzednio rozważanych przypadków, sieci nie uda się ich rozpoznać, co oznaczać będzie sygnał ostrzegawczy, że są one odmienne od wszystkich znanych do tej pory klas. W ten sposób nauczona sieć Kohonena może służyć jako detektor sygnalizujący pojawienie się nowości.

Uogólniając powyższe rozważania należy stwierdzić, że sieć Kohonena dokonuje grupowania obiektów, opisuje zależności pomiędzy tymi grupami oraz pozwala na zaklasyfikowanie nowych obiektów do wcześniej rozpoznanych klas. Jest to więc bogate w możliwości, wysoce użyteczne narzędzie mogące służyć do eksploracyjnej analizy danych.

6.2. Sieć neuronowa Kohonena jako narzędzie wstępnego grupowania wektorów wartości wejściowych.

W niniejszym podrozdziale przedstawiona zostanie bardzo obiecująca strategia decyzyjna o jakościowo odmiennym podejściu gdyż bazująca na sieci Kohonena wykorzystanej jako narzędzie wstępnego grupowania wektorów wartości wejściowych.

Główna idea metody jest następująca:

- W oparciu o ciąg uczący (obejmujący dane wejściowe analogiczne jak w poprzednich rozdziałach) realizowany jest proces grupowania wzorców (wektorów), opisujących stan rynku w poszczególnych chwilach czasowych (sesjach giełdowych). Po zakończeniu procesu uczenia sieci SOM (grupowania) pewne neurony mapy reprezentują związane z nimi skupienia (klastry) wzorców.
- Na tej podstawie dokonywana jest analiza przyszłych stóp zwrotu uzyskanych dla wzorców należących do każdej grupy osobno, analiza ta obejmuje obliczenie średniej przyszłej stopy zwrotu dla wzorców z danej klasy oraz obliczenie pewnego przyjętego wskaźnika będącego miarą ryzyka inwestycyjnego. Wskaźnikiem takim może być np. odchylenie standardowe przyszłych stóp zwrotu dla rozważanej klasy.

- Na podstawie analizy otrzymanych dla każdej klasy wskaźników zysku i ryzyka, a także w oparciu o indywidualne preferencje inwestora (dotyczące np. poziomu akceptowalnego ryzyka), z każdą grupą wzorców kojarzona jest określona decyzja inwestycyjna, np. kup, sprzedaj, utrzymuj pozycję inwestycyjną.
- W trybie realizacji (wykorzystania) skonstruowanego modelu, zakwalifikowanie pojawiającego się nowego wzorca opisującego bieżącą dynamikę rynku danego waloru do określonej klasy jednoznacznie determinuje decyzję inwestycyjną. Model taki jest więc funkcjonalnym modelem decyzyjnym realizującym określoną strategię inwestycyjną.

Poniżej przedstawiono wyniki badań efektywności modelu decyzyjnego zbudowanego według zaproponowanej metody, w zagadnieniu realizacji strategii inwestycyjnej dla indeksu WIG20 i kontraktów terminowych na indeks w okresie od 23.02.2000r. do 25.11.2005r, co stanowi 1444 sesji giełdowych. Zadaniem modelu było generowanie decyzji inwestycyjnej (na podstawie analizy bieżącej sytuacji rynkowej) dla okresu od otwarcia notowań w dniu bieżącym (moment realizacji decyzji) do otwarcia notowań podczas kolejnej sesji giełdowej (kiedy może zostać zrealizowana następna transakcja). W trakcie realizacji aktywnej strategii inwestycyjnej założono możliwość realizacji dwóch typów transakcji: kupna lub krótkiej sprzedaży kontraktu terminowego na indeks WIG20.

Dane wejściowe zostały podzielone na dwa zbiory: zbiór uczący w skład którego weszło 1144 wzorców oraz zbiór walidacyjny z 300 przypadkami.

Dobierając parametry struktury i uczenia sieci SOM zostały przeanalizowane 3 warianty sieci - zawierające 4, 9 i 16 neuronów na dwuwymiarowej mapie o wymiarach 2×2 , 3×3 , 4×4 , euklidesową metrykę określającą odległość między wektorami w przestrzeni cech, oraz współczynnik uczenia równy 0,6 i malejący wraz z upływem czasu uczenia. Przyjęto 1000 epok uczenia. Rozmiar sąsiedztwa ustalono na początku procesu uczenia na maksymalny (obejmujący wszystkie neurony), ale malejący liniowo podczas uczenia do zera.

Po przeprowadzeniu procesu uczenia sieci zaobserwowano, iż nie wystąpiła sytuacja, aby dla któregoś z wektorów danych wejściowych dany neuron nie stał się neuronem zwycięskim. Przyjęto zatem, iż każdy neuron reprezentuje osobne skupienie (grupę) wzorców (choć sąsiednie obszary mapy mogą odwzorowywać podobne wzorce), a zatem liczba tych grup jest równa w zależności od wariantu: 4, 9, 16. Obliczone dla zbioru uczącego istotne

wskaźniki charakteryzujące poszczególne grupy, takie jak: liczba wzorców n , średnia przyszła logarymiczna jednodniowa stopa zwrotu r (pomnożona przez 1000), oraz odchylenie standardowe dla tych stóp zwrotu σ (pomnożone przez 1000) przedstawiono w poniższych tabelach.

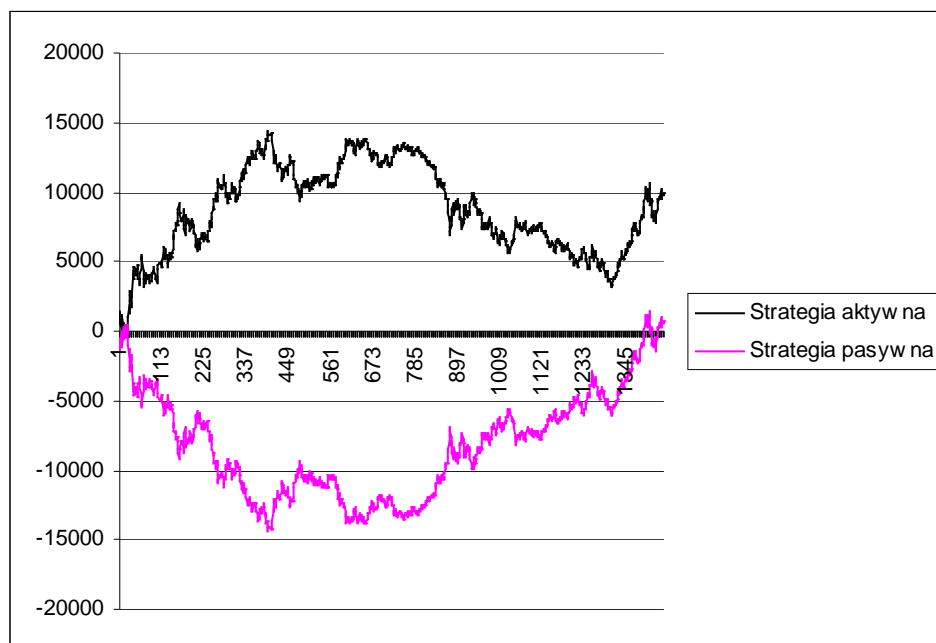
W celu powiązania z każdą grupą wzorców określonej decyzji inwestycyjnej przyjęto, iż podstawę tej decyzji stanowić będzie wartość wskaźnika s przedstawiająca stosunek stopy zwrotu do ryzyka dla wzorców danej grupy (miarą ryzyka jest odchylenie standardowe stóp zwrotu). W istocie s to powszechnie używany w analizie finansowej wskaźnik Sharpe'a przy założeniu, iż stopa procentowa aktywów wolnych od ryzyka wynosi zero. Wskaźnik ten jest często wykorzystywaną przez inwestorów miarą efektywności inwestycji w dany walor, uwzględnia bowiem powiązane ze sobą dwa podstawowe czynniki decyzyjne: oczekiwany zysk oraz ryzyko. W niniejszych badaniach założono, iż jeśli wartość bezwzględna tego wskaźnika przekracza określoną wielkość (przyjęto 5%), stanowi to podstawę do podjęcia decyzji kupna (gdy $s > 0,05$) lub krótkiej sprzedaży waloru (gdy $s < -0,05$); w przeciwnym przypadku należy utrzymać bieżącą pozycję inwestycyjną bez zmian. Tą regułą decyzyjną można więc wyrazić formułą:

- Jeżeli $s > 5\%$ (a akcje są krótko sprzedane) to dokonaj kupna akcji (K),
- Jeżeli $s < -5\%$ (a akcje są kupione) to dokonaj krótkiej sprzedaży akcji (S),
- W przeciwnym przypadku utrzymaj bieżącą pozycję inwestycyjną bez zmian (T).

Wartości współczynnika s (dla danych uczących), oraz związane z nimi decyzje inwestycyjne przyporządkowane do poszczególnych klas wzorców zidentyfikowanych przez sieć SOM, również umieszczono w poniższych tabelach.

Nr grupy (nr neuronu)	Liczba wzorców n	Śr. log. stopa zwrotu $r \times 1000$	Odch.std. $\sigma \times 1000$	$s = r / \sigma$	Decyzja
1	172	3,0	28,8	10,35%	K
2	622	0,2	21,9	0,98%	T
3	253	-3,8	37,2	-10,15%	S
4	397	1,0	22,9	4,20%	T

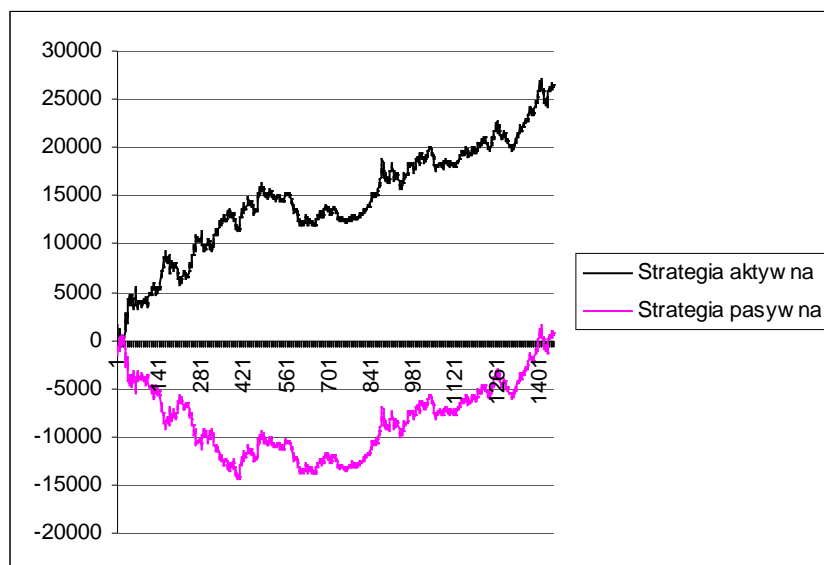
Tabela 27. Wartości podstawowych wskaźników oraz wynikających z nich decyzji inwestycyjnych dla poszczególnych grup wzorców dla sieci Kohonena z 4 neuronami w warstwie drugiej.



Rys. 17. Wartość portfela przy zastosowaniu dwóch strategii: aktywnej i pasywnej w przypadku hipotetycznej inwestycji w indeks WIG20 przy pomocy sieci SOM z 4 neuronami, bez uwzględnienia prowizji maklerskiej od zawieranych transakcji kupna/sprzedaży.

Nr grupy (nr neuronu)	Liczba wzorców n	Śr. log. stopa zwrotu $r \times 1000$	Odch.std. $\sigma \times 1000$	$s = r / \sigma$	Decyzja
1	108	-2,5	29,0	-8,65%	S
2	110	-2,6	30,9	-8,49%	S
3	36	-12,0	65,0	-18,47%	S
4	159	3,0	20,4	14,68%	K
5	292	1,1	24,0	4,68%	T
6	88	3,5	32,8	10,76%	K
7	264	-1,1	24,2	-4,45%	T
8	198	-0,1	19,4	-0,69%	T
9	189	1,4	21,3	6,50%	K

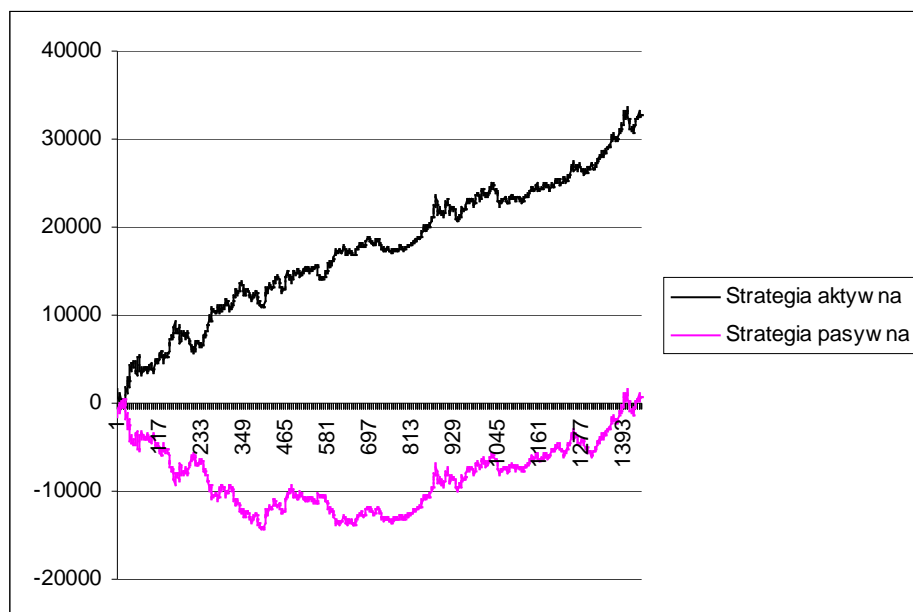
Tabela 28. Wartości podstawowych wskaźników oraz wynikających z nich decyzji inwestycyjnych dla poszczególnych grup wzorców dla sieci Kohonena z 9 neuronami.



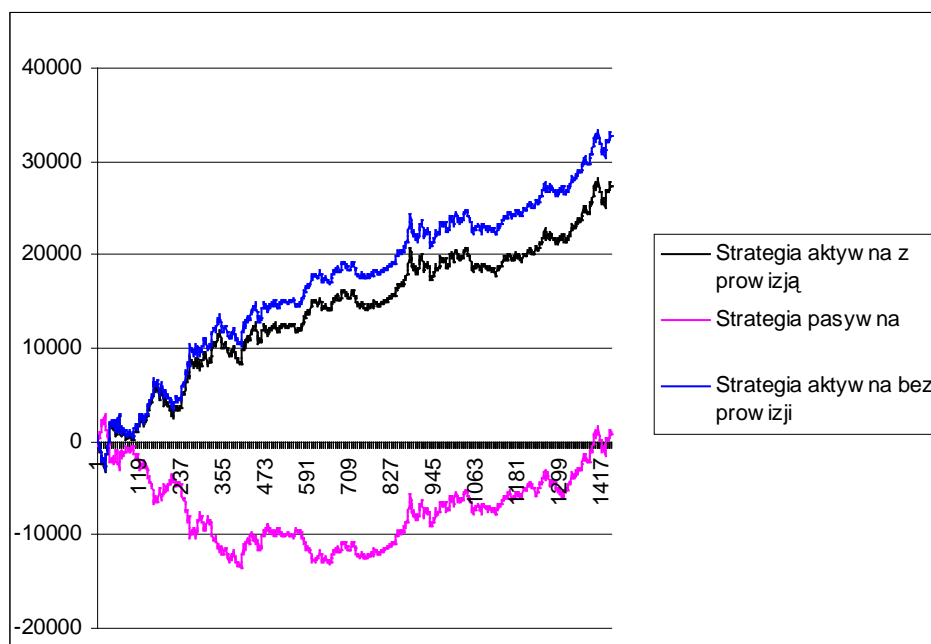
Rys. 18. Wartość portfela przy zastosowaniu dwóch strategii: aktywnej i pasywnej w przypadku hipotetycznej inwestycji w indeks WIG20 przy pomocy sieci SOM z 9 neuronami, bez uwzględnienia prowizji maklerskiej od zawieranych transakcji kupna/sprzedaży.

Nr grupy (nr neuronu)	Liczba wzorców n	Śr. log. stopa zwrotu $r \times 1000$	Odch.std. $\sigma \times 1000$	$s = r / \sigma$	Decyzja
1	36	-12,0	65,0	-18,47%	S
2	51	-2,9	27,0	-10,80%	S
3	115	2,7	21,8	12,22%	K
4	83	3,6	33,4	10,72%	K
5	108	-2,1	30,9	-6,87%	S
6	60	-2,7	30,5	-8,98%	S
7	104	1,2	17,9	6,75%	K
8	62	-1,6	20,4	-7,97%	S
9	161	-0,5	23,4	-2,26%	T
10	55	5,7	30,7	18,54%	K
11	55	1,7	34,0	4,89%	T
12	108	1,9	20,2	9,34%	K
13	159	-1,0	20,0	-4,83%	T
14	139	1,3	14,5	8,66%	K
15	90	-3,7	23,2	-16,10%	S
16	58	3,7	27,9	13,17%	K

Tabela 29. Wartości podstawowych wskaźników oraz wynikających z nich decyzji inwestycyjnych dla poszczególnych grup wzorców dla sieci Kohonena z 16 neuronami.



Rys. 19. Wartość portfela przy zastosowaniu dwóch strategii: aktywnej i pasywnej w przypadku hipotetycznej inwestycji w indeks WIG20 przy pomocy sieci SOM z 16 neuronami, bez uwzględnienia prowizji maklerskiej od zawieranych transakcji kupna/sprzedaży.



Rys. 20. Wartość portfela w przypadku rzeczywistej inwestycji w kontrakty terminowe na indeks WIG20 przy zastosowaniu trzech strategii inwestycyjnych: pasywnej, aktywnej opartej o decyzje analizowanego modelu przy zerowej prowizji maklerskiej, oraz aktywnej j.w. przy prowizji wynoszącej 30 PLN od zmiany zajmowanej pozycji.

Analizując powyższe zestawienia tabel i wykresów można dojść do dwóch zasadniczych wniosków. Po pierwsze w każdym z przedstawionych modeli (przy wykorzystaniu aktywnej neuronowej strategii decyzyjnej) zysk w rozważanym okresie czasu, jest istotnie większy od korzyści osiąganych przy wykorzystaniu strategii pasywnej, gdzie

zysk w przybliżeniu niewiele różni się od zera (734 PLN w sześcioletnim horyzoncie inwestycyjnym). Po drugie próba wykorzystania w tworzonych modelach sieci neuronowej Kohonena o zbyt małej liczbie neuronów w drugiej warstwie (w rozważanym powyżej modelu 4 neurony w dwuwymiarowej mapie 2 x 2) nie pozwala na realizację zadania wstępnego grupowania danych. Dopiero model z 9 i 16 neuronami sprostał postawionemu zadaniu. Najlepszym okazał się model zawierający 16 neuronów na dwuwymiarowej mapie o wymiarach 4 x 4. Hipotetyczny możliwy do osiągnięcia zysk na danych historycznych, bez uwzględnienia prowizji od zawieranych transakcji kupna/sprzedaży (Rys. 19) wyniósł w rozważanym okresie 32787 PLN, podczas gdy realny dochód (na podstawie inwestycji w kontrakty terminowe na indeks WIG20 – rysunek 20) już uwzględniając prowizję maklerską w wysokości 30 PLN od przeprowadzonej transakcji wyniósł w tym samym czasie 27440 PLN.

W tym miejscu czytelnika ciekawi zapewne jakie zyski można osiągnąć stosując powyższą strategię decyzyjną dla danych najnowszych, oraz jak wygląda porównanie tych zysków z zyskiem osiągniętym w przypadku inwestycji pasywnej (kupieniu na początku i sprzedaniu na końcu rozważanego okresu czasu). Takie zestawienie realnego zysku w okresie od 29-11-2005 do 20-04-2006 przedstawia poniższa tabela.

	Inwestycja w WIG20		Inwestycja w kontrakty	
	Bez uwzględnienia prowizji	Po odjęciu prowizji	Bez uwzględnienia prowizji	Po odjęciu prowizji
Strategia aktywna	5828	5798	5650	5620
Strategia pasywna	5828	5798	5650	5620

Tabela 30. Zestawienie zysków osiągniętych przy pomocy neuronowej strategii decyzyjnej dla danych najnowszych z okresu 29-11-2005 do 20-04-2006.

Jak widać z zestawienia przedstawionego w tabeli 30 wartości zysku zarówno dla strategii aktywnej jak i strategii pasywnej są jednakowe. Dzieje się tak dlatego, ponieważ sieć neuronowa Kohonena dokonując wstępnego grupowania danych dokonuje tego de facto w postaci średnioterminowego trendu wygładzając i pomijając większość krótkoterminowych wahań. A zatem testując strategię inwestycyjną uzyskaną przy pomocy analizowanych sieci neuronowych SOM dla danych z okresu 29-11-2005 do 20-04-2006 uzyskano dla całego okresu czasu jednakową strategię, z której wynika, iż należy w perspektywie 100 sesji

giełdowych utrzymywać długą pozycję inwestycyjną, co jest w tym wypadku jednoznacznie ze strategią pasywną.

6.3. Podsumowanie.

Wykorzystanie grupowania danych przy pomocy samoorganizujących sieci neuronowych typu Kohonena do generowania finansowych strategii decyzyjnych na Giełdzie Papierów Wartościowych to nowatorskie podejście, które jak zostało pokazane przynosi całkiem niezłe rezultaty, a otrzymane wyniki świadczą o wysokiej efektywności rozważanej metody.

Porównanie tak skonstruowanych modeli decyzyjnych opartych na sieciach neuronowych typu SOM z modelami przedstawionymi w rozdziale 4, opartymi na prognozie wartości indeksu WIG20 pokazuje wyższość tych pierwszych. Okazuje się bowiem, że w przypadku strategii opartych na samoorganizujących sieciach nie występuje problem opóźnienia czasowego, które może mieć bardzo istotny wpływ na efektywność modelu. Ponadto w rozważanym przypadku sieci Kohonena „odgadły” średnioterminowy trend, co w znaczący sposób redukuje ilość zawieranych transakcji, a przez to również koszty prowizji maklerskiej. Uzyskane jednak w ten sposób wyniki są gorsze od wyników uzyskanych w rozdziale 5 (co było do przewidzenia), gdzie przy wykorzystaniu głównie sieci neuronowych typu RBF tworzone były strategie inwestycyjne i generowane sygnały kupna/sprzedaży.

Jako dodatkową zaletę przedstawionej w niniejszym rozdziale metody zaliczyć można elastyczność dostosowania parametrów strategii decyzyjnej modelu do indywidualnych preferencji inwestora (np. dotyczących ryzyka inwestycyjnego).

7. Zakończenie

W pracy została zaprezentowana idea wspomaganego neuronalnie modelu decyzyjnego. Po próbie (w rozdziale 3) optymalnego doboru wektora zmiennych wejściowych przy pomocy różnych metod dostępnych w pakiecie Statistica, dokonano budowy dwóch oddzielnych grup modeli służących do prognozowania zjawisk ekonomicznych na Giełdzie Papierów Wartościowych w Warszawie. W rozdziale 4 skupiono się na predykcji procesów gospodarczych w postaci neuronowego prognozowania wartości indeksu WIG20 i próbie budowy systemu decyzyjnego na tak wyestymowanych wartościach indeksu.

Jednak, co zostało pokazane w niniejszej pracy, sama tylko predykcja procesów gospodarczych nie jest najwygodniejszą ani najkorzystniejszą formą komputerowego wspomaganie ekonomicznych procesów decyzyjnych. Wynika to z faktu, że w ostatnim czasie mamy do czynienia z prawdziwym zalewem informacji pochodzących z różnych źródeł. Obecnie problemem nie jest brak informacji czy pytanie skąd je zdobyć, powstaje natomiast problem ich szybkiego i efektywnego przetwarzania oraz selekcji. Przedstawiony w rozdziale 3 dobór zmiennych wejściowych do wszystkich rozważanych w niniejszej pracy modeli jest tylko zarysem problemu wyboru optymalnego wektora wejściowego. Wobec ogromu dostępnych danych gospodarczych i ekonomicznych wybór tych właściwych opiera się niejednokrotnie na intuicji i subiektywnej ocenie badacza, a pominięcie istotnej zmiennej może być przyczyną znaczącej różnicy w efektywności opracowanego modelu. W związku z powyższym sam wybór optymalnego wektora zmiennych wejściowych może stanowić problematykę dalszych badań i analiz.

Rozdział 5 i 6 niniejszej pracy stanowi dowód postawionych na wstępie dwóch tez. W rozdziale 5 wykazano, poprzez porównanie dwóch klas modeli (pierwsza grupa to tradycyjne modele prognostyczne, których zadaniem było prognozowanie zmiennej wyjściowej - wartości indeksu o jeden lub dwa okresy naprzód, druga grupa natomiast to neuronowe modele decyzyjne, których zadaniem było przewidywanie kierunku zmian czy występującego trendu), że rozwiązanie zmierzające wprost do formułowania przez neuronowy model sugestii decyzji inwestycyjnych (lub gotowych sygnałów kupna-sprzedaży) ma przewagę nad podejściem opartym na odrębnym budowaniu prognoz i osobnym wspomaganie procesów informacyjnych zmierzających do decyzji – nie wspominając o tym,

że pierwsze podejście, jest o wiele wygodniejsze dla użytkowników. W tym celu użyto dwóch rodzajów sieci neuronowych: MLP oraz RBF.

Rozdział 6 stanowi dowód drugiej, bardziej szczegółowej tezy. Uzyskane przy pomocy samoorganizujących sieci neuronowych Kohonena wyniki są co prawda gorsze od modeli z rozdziału 5, jednak ich efektywność w porównaniu z modelami z rozdziału 4 jest znacząco lepsza. Dodatkową zaletą przedstawionej metody może być elastyczność dostosowywania parametrów strategii decyzyjnej modelu do indywidualnych preferencji inwestora (np. dotyczących ryzyka inwestycyjnego). Z kolei jako kierunek dalszych badań (stanowiących potencjalną tematykę kolejnych opracowań) należy wskazać kwestię doboru parametrów sieci SOM.

Rozważania niniejszej rozprawy kończy dodatek, w którym przedstawiono krótki przegląd różnych metod prognozowania procesów ekonomicznych, które można znaleźć we współczesnej literaturze. Jakkolwiek przegląd ten jest zapewne niepełny i niekompletny, to jednak pokazuje on, że na temat sposobów przewidywania przyszłych wartości takich czy innych ekonomicznych szeregów czasowych napisano już bardzo wiele wartościowych prac.

Za najważniejsze oryginalne elementy zawarte w niniejszej pracy uznać należy:

- Porównanie strategii decyzyjnych (inwestycyjnych) budowanych na podstawie prognozy wartości indeksu WIG20 uzyskanej przy pomocy sieci neuronowych z doradczym systemem inwestycyjnym opartym na neuronowym modelu decyzyjnym. Takie porównanie pozwoliło udowodnić, że bezpośrednie tworzenie strategii inwestycyjnych przynosi znacząco lepsze rezultaty niż opieranie się w tworzonych modelach na prognozie wartości indeksu.
- Wykorzystanie do budowy doradczego systemu inwestycyjnego samoorganizujących sieci neuronowych typu Kohonena i pokazanie, że tak stworzone modele decyzyjne mogą być bardziej przydatne niż tworzenie strategii inwestycyjnych opartych na prognozie wartości indeksu. To podejście ma dodatkowo jeszcze jedną zaletę, mianowicie pozwala potencjalnemu inwestorowi na elastyczne dostosowanie parametrów strategii decyzyjnej modelu do indywidualnych preferencji (np. dotyczących ryzyka inwestycyjnego).
- W pracy ponadto pokazano różne metody optymalnego doboru wektora zmiennych wejściowych do modeli neuronowych przy pomocy pakietu Statistica. Pozwala to na optymalny dobór zmiennych spośród wstępnie wybranych przez badacza (np. w sposób subiektywny) danych wejściowych.

Dodatek A

Metody analizy i prognozowania finansowych szeregów czasowych.

Szereg czasowy to realizacja procesu stochastycznego, którego dziedziną jest czas. Inaczej jest to zbiór informacji uporządkowanych w czasie, których pomiary wykonywane są z dokładnym krokiem czasowym.

Głównymi celami analizy szeregów czasowych jest: wykrywanie natury zjawiska reprezentowanego przez sekwencję obserwacji i prognozowanie (przewidywanie przyszłych wartości szeregu czasowego). Oba wymagają zidentyfikowania i opisanie, w sposób mniej lub bardziej formalny, elementów szeregu czasowego, a raz ustalony wzorzec może zostać zastosowany do innych danych (w obrębie tego samego modelu). Niezależnie od trafności teoretycznego uzasadnienia postaci modelu, zawsze możemy przewidywać przyszłe wartości szeregu czasowego na drodze ekstrapolacji.

Procesem stochastycznym jest zbiór zachowań pewnego układu, zależny od upływu czasu oraz od występowania zjawisk losowych, przy czym czas w tym układzie jest zmienną niezależną. W praktyce przyjmuje się definicję, że proces stochastyczny jest to zjawisko (reprezentowane np. poprzez szereg czasowy), w którym przyszła wartość opisująca stan zjawiska nie pojawi się z prawdopodobieństwem równym 1.

Już w 1906 roku znany rosyjski matematyk A.A. Markow (1856 – 1922) rozpoczął badania nad zdarzeniami zależnymi, spośród których wyodrębnił schemat doświadczeń połączonych w łańcuch bez tzw. efektów następstwa. Wyniki prac Markowa doprowadziły do rozwoju klasy procesów stochastycznych posiadających własność braku pamięci zwanych procesami Markowa [25][44][71]. Proces Markowa jest definiowany jako proces stochastyczny, w którym prawdopodobieństwo każdego zdarzenia zależy jedynie od wyniku poprzedniego.

Klasycznym przykładem niech będzie zjawisko, w którym prawdopodobieństwo, że jutrzejsza zmiana (cen akcji, wartości indeksu, zmiennej ekonomicznej itp.) będzie miała ten sam znak co dzisiejsza jest większe, niż to, że znak będzie przeciwny. Tak określony proces jest ciągiem Markowa pierwszego rzędu. W przypadku analizy portfelowej [22][83], gdzie inwestycje dobiera się w taki sposób, by przy założonej stopie zysku ponieść jak najmniejsze ryzyko mamy do czynienia z ciągiem Markowa zerowego rzędu, czyli bez względu na to jakie było historyczne zachowanie wartości to przyszłe zmiany będą określone takimi samymi związkami prawdopodobieństw. Podobna sytuacja występuje w przypadku teorii rynku efektywnego, mówiącej o tym, że wszystkie dotychczasowe zdarzenia są już odzwierciedlone

w cenach akcji. Zatem zwolennicy efektywności rynku są tak naprawdę zwolennikami tezy, że zachowanie akcji na giełdzie jest procesem Markowa zerowego rzędu.

Zupełnie inaczej sytuacja wygląda w przypadku zwolenników analizy technicznej, która bazuje na klasycznych wskaźnikach obliczanych na podstawie wcześniejszych wartości procesu (cen akcji, wartości indeksu). W takim wypadku zmiana cen akcji na giełdzie będzie procesem Markowa wysokiego rzędu (przykładowo często stosowany w analizie technicznej wskaźnik RSI jest obliczany na podstawie wartości historycznej z 14 okresów). Należy pamiętać również, że proces Markowa bazuje wyłącznie na rozkładzie prawdopodobieństw warunkowych. Może się więc zdarzyć, że mamy do czynienia z chaosem deterministycznym, w którym jutrzejsze zachowanie określone jest wzorem matematycznym, a mimo to proces będzie sprawiał wrażenie, że jest zerowego rzędu, czyli niezależne od przeszłości. Wynika to z faktu, że bardzo podobne, niemal identyczne zachowanie historyczne może skutkować zupełnie różnym zachowaniem w przyszłości (tzw. efekt motyla). Bardzo ciekawe omówienie zasad inwestowania w oparciu o teorię chaosu przedstawia książka: [65]. Istnieje wiele nieliniowych, deterministycznych modeli ekonomicznych, w których pojawiają się rozwiązania chaotyczne [49]. Dotychczas jednak nie znaleziono przekonujących dowodów na istnienie chaosu deterministycznego w empirycznych szeregach czasowych, opisujących dynamikę układów socjoekonomicznych. Prawdopodobnie powodem tego jest nakładanie się tego efektu z szumem losowym, co jest bardzo trudne do rozseparowania, jeśli ciąg danych jest krótki lub system jest niestacjonarny. Pierwsza metoda kontroli chaosu została zaproponowana w 1990 r. [63], a możliwość użycia kontroli chaosu do sterowania sprzedażą za pomocą małych, ale odpowiednio dobranych zmian parametru inwestycji w chaotycznym modelu mikroekonomicznym wykazano w pracach: [33][34].

Wracając do stochastycznych procesów Markowa można zaklasyfikować je w zależności od tego czy oś czasu i zbiór możliwych wartości procesu są zbiorami dyskretnymi czy ciągłymi. W przypadku procesów ciągłych na szczególną uwagę zasługują procesy dyfuzji, których nazwa wywodzi się z fizyki i oznacza zjawisko samorzutnego mieszania się różnych substancji we wszystkich stanach skupienia. W 1827 roku brytyjski biolog Robert Brown obserwując pyłki kwiatowe w zawieszynie wodnej dostrzegł, iż znajdują się one w nieustannym, chaotycznym ruchu. To zjawisko nazywane jest ruchem Browna. Autorami matematycznego opisu ruchów Browna byli niezależnie Albert Einstein [20] oraz Marian Smoluchowski [70]. Obaj naukowcy zauważyli, że przypadkowe błądzenie pyłków jest wywołane bombardowaniem ich przez cząsteczki wody. Cząsteczki wody są dużo mniejsze, jest ich wiele oraz poruszają się bardzo szybko. Różnice w prędkości ruchu oraz

ilości uderzających cząsteczek z poszczególnych stron są przyczyną ruchów drobin pyłku w cieczy. Nad ruchami Browna pracował również w późniejszym okresie N. Wiener [82]. Pracom Einsteina i Smoluchowskiego na temat ruchów Browna zawdzięczamy wprowadzenie do fizyki procesów stochastycznych opisujących układy losowe, w których rozkłady prawdopodobieństwa zmieniają się w czasie. Interesujące jest, że procesy stochastyczne zostały wprowadzone wcześniej do ekonomii niż do fizyki, bowiem stało się to już w roku 1900 dzięki Louisowi Bachalierowi, który zaproponował model przypadkowego błędzenia dla opisu fluktuacji kursów akcji na giełdzie [80].

Obecnie jednym z podstawowych narzędzi tzw. inżynierii finansowej jest teoria Blacka-Scholesa, nagrodzona Nagrodą Nobla z ekonomii w roku 1997, a opracowana przez absolwenta fizyki i doktora matematyki Fischera Blacka oraz ekonomistę Myrona Scholesa [37]. Teoria Blacka-Scholesa pozwala na wycenę wartości finansowych instrumentów pochodnych (opcji), oraz służy do optymalizacji „bezpiecznego” portfela inwestycyjnego. Wykorzystywane są tutaj znane własności stochastycznych równań różniczkowych opartych na procesie Wienera, który jest podstawą w modelowaniu szumów dla układów fizycznych [26]. Popularność teorii Blacka-Scholesa w dużym stopniu przyczyniła się do upowszechnienia stosowania procesów dyfuzji w zagadnieniach ekonomicznych, m.in. w zagadnieniach zarządzania ryzykiem, konstrukcji optymalnych strategii inwestycyjnych czy wyceny instrumentów pochodnych [13][15][17][24][36][38][40][41][42][47][60][80][84]. Rezultaty teorii Blacka-Scholesa są jednak w ostatnich latach często podważane, bowiem teoria ta nie obejmuje „zjawiska” gwałtownych załamań kursów na giełdach. Stworzenie opisu dla tego zagadnienia (i przewidzenia momentów takich załamań) jest obecnie z oczywistych powodów w centrum zainteresowania specjalistów w zakresie rynków kapitałowych. Podejmowane próby opierają się między innymi na teorii zjawisk krytycznych [72], teorii skalowania, multifraktali i niegaussowskich procesów stochastycznych [53] oraz teorii przejść Kramera [9].

Prognozowanie na podstawie najważniejszych modeli:

Poniżej zostanie dokonany przegląd technik, które są przydatne do analizy danych szeregów czasowych.

Na analizę szeregów czasowych (co zostało już powiedziane na początku niniejszego dodatku) składa się: wykrywanie natury zjawiska i prognozowanie przyszłych wartości szeregu czasowego. Tak jak w większości innych analiz, w analizie szeregów czasowych zakłada się, że w danych można wyodrębnić składnik systematyczny oraz losowy szum

(zakłócenia), który utrudnia identyfikację struktury zjawiska. Większość technik analizy szeregów czasowych wiąże się z pewnymi formami filtrowania szumu w celu uwidocznienia składnika systematycznego, a strukturę większości szeregów czasowych można opisać przy pomocy dwóch podstawowych klas składników: trendu i sezonowości (patrz [11][78]). Niestety takie podejście wiąże się z pewnym problemem, bowiem wymaga od badacza wiedzy na temat matematycznego modelu procesu. Jednak w rzeczywistych badaniach struktura szeregu jest często niewyraźna, wariancja składnika losowego jest znaczna, a my mimo to potrzebujemy nie tylko odkryć ukryty wzorec, ale także generować prognozy. W tym zagadnieniu pomocna jest metodologia ARIMA, która została rozwinięta przez Boxa i Jenkinsa [11]. Pomimo swych zalet, mocy i elastyczności [32][64][76], ARIMA jest techniką złożoną (nie jest łatwa w użyciu, wymaga dużego doświadczenia i chociaż często daje zadowalające wyniki, to zależą one w dużym stopniu od badacza [4]). W modelu wyróżnia się trzy typy parametrów: parametry autoregresyjne (p), rząd różnicowania (d), oraz parametry średniej ruchomej (q). Do identyfikacji modelu należy zidentyfikować liczbę i typ tych parametrów. Podstawowym narzędziem tej identyfikacji są wykresy szeregów, korelogramy autokorelacji i autokorelacji cząstkowej. Jednak model ARIMA ma zastosowanie tylko do szeregów stacjonarnych to znaczy, powinien on mieć stałą w czasie średnią, wariancję i autokorelację, a w praktyce większość szeregów czasowych to szeregi niestacjonarne o zmiennej wartości oczekiwanej. W takim przypadku należy dokonać albo ekstrakcji trendu albo wykonać różnicowanie aż do osiągnięcia stacjonarności.

Model wyrównywania wykładniczego.

Wyrównywanie wykładnicze stało się bardzo popularne jako metoda prognozowania wielu typów szeregów czasowych. Metoda ta została rozwinięta niezależnie przez Browna i Holta.

Jednolitą klasyfikację metod wyrównywania wykładniczego zaproponował Gardner [27]. Znakomite wprowadzenie można także znaleźć w: [52][56].

Wzór dla prostego wyrównywania wykładniczego wygląda następująco: $S_t = \alpha * X_t + (1 - \alpha) * S_{t-1}$, gdzie: X_t - oznacza zaobserwowane wartości szeregu, a S_t - wartości wygładzone. Zgodnie z rekurencyjną procedurą każda nowa wygładzona wartość jest obliczana jako średnia ważona bieżącej obserwacji i poprzedniej obserwacji wygładzonej. Zatem w konsekwencji, każda wartość wygładzona jest ważoną średnią poprzednich obserwacji, przy czym wagi maleją wykładniczo w zależności od wartości parametru α .

Chociaż wykonano sporo badań na temat teoretycznych własności (prostego i złożonego) wyrównywania wykładniczego (np. patrz [27][61]), to metoda ta zyskała popularność głównie z powodu swojej użyteczności jako narzędzia prognoz. Na przykład badania empiryczne wykonane przez Makridakisa i in. [52], pokazały, że proste wyrównywanie wykładnicze dało najlepsze prognozy na jeden okres naprzód spośród 24 innych metod analizy szeregów czasowych i po zastosowaniu rozmaitych miar dokładności prognozy. Zatem, bez względu na model teoretyczny procesu kryjącego się za obserwowanym szeregiem czasowym, proste wyrównywanie wykładnicze dostarcza często całkiem trafnych prognoz.

Analiza widmowa:

Analiza widmowa służy do badania struktury harmonicznej szeregu czasowego. Celem tej analizy jest dekompozycja złożonego szeregu czasowego zawierającego składniki cykliczne na kilka podstawowych funkcji sinusoidalnych (sinus i cosinus) o określonych długościach fali. Pojęcie „widmo” jest odpowiednią metaforą natury tej analizy: załóżmy, że studiujemy wiązkę białego światła słonecznego, która wygląda na początku jak losowe (biały szum) nagromadzenie światła o różnych długościach fal. Kiedy jednak przepuścimy światło przez pryzmat, możemy odseparować różne długości fal lub składniki cykliczne, z których składa się białe światło słoneczne. Faktycznie, wykorzystując tę technikę możemy teraz zidentyfikować i wydzielić różne źródła światła. Zatem, identyfikując ważne podstawowe składniki cykliczne, dowiedzieliśmy się czegoś na temat interesującego nas zjawiska. W istocie, wykonanie analizy widma szeregu czasowego przypomina przepuszczenie szeregu przez pryzmat w celu identyfikacji długości fal i znaczenia podstawowych składników cyklicznych. W wyniku analizy można odkryć w interesującym nas szeregu czasowym kilka cykli okresowych o różnych długościach, które na początku wyglądały na mniej lub bardziej losowy szum.

Często przytaczanym przykładem analizy widmowej jest cykliczna natura aktywności plam słonecznych (np. patrz [8][69]). Okazuje się, że aktywność plam słonecznych ulega zmianie w cyklach 11 letnich. W literaturze, w celu zilustrowania tej techniki przytacza się także inne przykłady zjawisk astronomicznych, meteorologicznych, wahań w cenach towarów, działalności ekonomicznej itd. W odróżnieniu od modeli ARIMA lub wyrównywania wykładniczego, celem analizy widmowej jest identyfikacja wahań okresowych o różnej długości, podczas gdy w przypadku tamtych analiz okres składnika cyklicznego jest zazwyczaj znana, a następnie włączona do pewnego modelu teoretycznego średnich ruchomych lub autokorelacji.

Klasyczną pozycją na temat analizy widmowej jest praca [8]; można także sięgnąć do innych szczegółowych rozważań: [12][14][21][43][66][69].

Przedstawione w niniejszym dodatku metody analizy oraz różne podejścia do szeregów czasowych, a w szczególności finansowych szeregów czasowych są tylko wycinkiem dorobku wielu badaczy. Ostatnio coraz większą popularnością cieszą się metody z pogranicza statystyki, ekonometrii i sztucznej inteligencji takich jak algorytmy genetyczne, stosowane w powyższej rozprawie sztuczne sieci neuronowe, czy wreszcie różne metody hybrydowe łączące klasyczne metody statystyczne z różnymi technikami heurystycznymi.

Dodatek B

Istota giełdowego kontraktu terminowego, należącego do instrumentów pochodnych notowanych na Giełdzie Papierów Wartościowych w Warszawie.

Instrumenty pochodne są instrumentami finansowymi, których cena bezpośrednio lub pośrednio zależy od ceny papierów wartościowych (akcje, obligacje) lub innych instrumentów takich jak kursy walut czy indeksy giełdowe.

Na GPW w Warszawie notowane są instrumenty pochodne obok instrumentów rynku kasowego. Obrót instrumentami pochodnymi dokonywany jest na tzw. rynku terminowym, w odróżnieniu od rynku kasowego, na którym handluje się rzeczywistymi aktywami, takimi jak akcje czy obligacje.

Na rynku kasowym transferowi podlega **kapitał**, natomiast na rynku terminowym **zobowiązania** (do przyszłej zapłaty lub dostarczenia instrumentu, który jest przedmiotem kontraktu). Zobowiązania te są nabywane przez strony transakcji terminowej w chwili zawarcia transakcji, a zapłata następuje w przyszłości, w przeciwieństwie do rynku kasowego, na którym zapłata następuje w chwili zawarcia transakcji.

Instrumenty pochodne są wykorzystywane przez inwestorów do trzech podstawowych celów:

Zabezpieczenia – zawierana transakcja ma na celu ograniczenie strat wynikających z inwestycji na rynku kasowym.

Spekulacji - zawierane są transakcje w celu osiągnięcia zysku wynikającego z przekonania o wzroście lub spadku wartości instrumentu pochodnego w przyszłości.

Arbitrażu - zawierane w celu osiągnięcia zysku bez ponoszenia ryzyka dzięki jednoczesnemu zawarciu transakcji na rynku kasowym i terminowym.

Do instrumentów pochodnych, które są notowane na GPW należą:

- kontrakty terminowe
- warranty
- jednostki indeksowe MiniWIG20
- certyfikaty indeksowe

Ze względu na tematykę niniejszej rozprawy bardziej szczegółowo opisana zostanie jedynie istota **kontraktów terminowych**.

Kontrakt terminowy (ang. *futures contract*) jest umową pomiędzy dwiema stronami, z których jedna zobowiązuje się do nabycia, a druga do sprzedaży w ściśle określonym, przyszłym terminie po ściśle określonej w momencie zawarcia transakcji cenie, określonej ilości wystandaryzowanego instrumentu bazowego, czyli instrumentu będącego przedmiotem transakcji. Kontrakty rozważanego tu typu zakładają również możliwość dokonania równoważnego rozliczenia finansowego. Kontrakty terminowe, typu *financial futures*, do których należy kontrakt na WIG20, nie opiewają na towar fizyczny, lecz na waluty, papiery wartościowe, depozyty bankowe o stałym oprocentowaniu, czy w końcu na jeszcze bardziej abstrakcyjne walory na przykład na indeksy giełdowe.

Jak wynika z powyższej definicji, w transakcji uczestniczy nabywca i sprzedawca kontraktu, których zapatrywania na przyszłą wartość instrumentu bazowego są odmienne.

Nabywca kontraktu otwierając **pozycję długą** (kupując kontrakt), zobowiązuje się i jednocześnie nabywa prawo do odebrania, kupienia, w przyszłości określonej ilości instrumentu bazowego po określonej cenie. Nabycie kontraktu stanowi więc zobowiązanie się do określonego działania w przyszłości. Nabywca kontraktu liczy, iż cena instrumentu bazowego w przyszłości będzie wyższa, od tej, po jakiej zobowiązał się go odebrać, nabywając kontrakt terminowy - i zrealizuje zysk wynikający z różnicy między ceną instrumentu bazowego w dniu wykonania kontraktu a ceną zadeklarowaną w kontrakcie. Nabywca kontraktu będzie miał bowiem prawo i obowiązek kupienia instrumentu bazowego po określonej cenie w dniu wygaśnięcia kontraktu. Jeżeli więc cena na rynku kasowym będzie wyższa, niż ta po jakiej inwestor zobowiązał się kupić instrument nabywając kontrakt, to w dniu wygaśnięcia kontraktu będzie mógł nabyć instrument bazowy po cenie niższej, wynikającej z kontraktu, i sprzedać drożej na rynku kasowym - realizując zysk. Jeżeli zaś cena na rynku kasowym będzie niższa od ustalonej w kontrakcie wówczas posiadacz długiej pozycji poniesie stratę.

Sprzedawca kontraktu zajmując **pozycję krótką** (sprzedając kontrakt), zobowiązuje się a zarazem nabywa prawo do dostarczenia, sprzedania, określonej ilości instrumentu na który opiewa kontrakt po określonej cenie w określonym momencie w przyszłości. Sprzedawca kontraktu podobnie jak nabywca zobowiązuje się do podjęcia ściśle określonego działania w przyszłym terminie. Sprzedawca kontraktu w odróżnieniu od nabywcy zakłada, że przyszła cena instrumentu bazowego będzie **niższa** od ceny po jakiej zobowiązał się dostarczyć instrument bazowy zgodnie z warunkami kontraktu. Jeżeli jego przewidywania się

sprawdzą i cena na rynku kasowym będzie niższa od ustalonej w kontrakcie, to posiadacz pozycji krótkiej będzie mógł kupić papiery taniej na rynku natychmiastowym i sprzedać drożej wykonując swoje zobowiązania wynikające ze sprzedaży kontraktu. Jeżeli natomiast cena instrumentu bazowego ustali się na poziomie wyższym od ustalonego w kontrakcie, klient wykonując swoje zobowiązanie poniesie stratę. Będzie zmuszony kupić instrument bazowy na rynku kasowym drożej i odsprzedać taniej wykonując swoje zobowiązanie wynikające z kontraktu.

Kontrakty terminowe występujące na GPW w Warszawie, ze względu na instrument bazowy można podzielić na:

1. Kontrakty terminowe indeksowe:
 - na indeks WIG20
 - na indeks TechWIG
 - na indeks MIDWIG
2. Kontrakty terminowe na akcje niektórych spółek
3. Kontrakty terminowe walutowe:
 - na kurs USD
 - na kurs EURO

Zajmiemy się teraz poruszonym w pracy kontraktem terminowym na indeks WIG20.

Kontrakt terminowy na indeks WIG20, obejmujący akcje dwudziestu największych spółek z warszawskiego parkietu został wprowadzony na początku 1998 roku przez Giełdę Papierów Wartościowych w Warszawie. Inwestorzy giełdowi otrzymali tym samym instrument, który umożliwia zarabianie zarówno na wzrostach, jak i na spadkach cen akcji na warszawskiej giełdzie. Dzięki prostej i przejrzystej budowie, kontrakty terminowe na WIG20 stały się w krótkim czasie instrumentem bardzo popularnym, o czym najlepiej świadczy dynamika wzrostu obrotów i liczba otwartych pozycji.

Konstrukcja kontraktu terminowego na WIG20:

- Nazwa skrócona: FW20kr,

gdzie:

F - oznaczenie kontraktu terminowego,

W20 - skrót nazwy instrumentu bazowego czyli w naszym przypadku WIG20,

k - kod określający miesiąc wykonania (H - marzec, M - czerwiec, U - wrzesień, Z – grudzień),

r - ostatnia cyfra roku wykonania,

Przykład: FW20U1- kontrakt dla którego instrumentem bazowym jest indeks WIG20, miesiąc wykonania przypada we wrześniu 2001 roku.

- Instrumentem bazowym jest indeks giełdowy WIG20.
- kontrakt jest rozliczany wyłącznie pieniężnie.
- Wartość kontraktu podawana jest w punktach indeksowych.
- Jeden punkt indeksowy odpowiada 10 PLN,
- Terminem realizacji, czyli wygaśnięcia kontraktu terminowego jest trzeci piątek - odpowiednio marca, czerwca, września lub grudnia,

Z kontraktami opisanego wyżej typu związana jest konieczność utrzymywania depozytu zabezpieczającego, którego wysokość ustala KDPW (w przypadku kontraktów terminowych wynosi on najczęściej 10-15% wartości kontraktu).

Jest to najbardziej istotna różnica między inwestowaniem w akcje, a inwestowaniem w kontrakty terminowe. Nie musimy bowiem w momencie zawierania transakcji angażować wszystkich środków, na które opiewają kontrakty. Z faktu pobierania od inwestorów depozytów zabezpieczających, a nie całości sumy, jaką jest wart kontrakt terminowy, wynika zasadnicza dla tego rynku cecha. Jest nią tak zwana „dźwignia finansowa”, która pozwala osiągać znacznie wyższe stopy zwrotu niż klasyczne inwestowanie w akcje. Drugą stroną medalu jest jednak fakt, że wszystkie instrumenty pochodne należą do najbardziej ryzykownych instrumentów finansowych.

Literatura

- [1]. Anderson J.A., Rosenfeld E., „Neurocomputing – Foundation of Research”, MIT Press, Cambridge, Mass., 1988.
- [2]. Azoff E.M., „Neural Networks Time Series Forecasting of Financial Markets”, Wiley, Chichester, 1994.
- [3]. Baestaens D.E., van den Bergh W.M., Wood D., „Neural Networks Solutions for Trading in Financial Markets”, Pitman Publishing, London, 1994.
- [4]. Bails D.G., Peppers L.C., „Business Fluctuations”, Englewood Cliffs, Prentice-Hall, 1982.
- [5]. Barr, D.S., Mani G.: „Using Neural Nets to Manage Investments”, AI Expert pp. 16-21, 1994.
- [6]. Beltratti A., Margarita S., Terna P.: „Neural Networks for Economic and Financial Modelling”, ITCP, London, 1996.
- [7]. Bishop, C.M., „Neural Networks for Pattern Recognition”, Oxford University Press, 1995.
- [8]. Bloomfield P., „Fourier Analysis of Time Series”, John Wiley and Sons, 1976.
- [9]. Bouchaud J., Cizeau P., Laloux L., Potters M., „Mutual Attractions: Physics and Finance”, Physics World, January 1999.
- [10]. Bouland H., Kamp Y., „Auto-association by multilayer perceptrons and singular value decomposition”, Biological Cybernetics No 59, pp. 291-294, 1988.
- [11]. Box, G.E.P., Jenkins, G., „Time Series Analysis: Forecasting and Control”, Holden-Day, 1976.
- [12]. Brigham E.O., „The Fast Fourier Transform”, Prentice-Hall, 1974.
- [13]. Brigo D., Mercurio F., „Interest Rate Models Theory and Practice”, Springer-Verlag, Berlin – Heidelberg – New York, 2001.
- [14]. Brillinger D.R., „Time series. Data analysis and theory” , Holt, Rinehart & Winston (1975).
- [15]. Butler C., „Tajniki Value at Risk”, Wydawnictwo K. E. Liber, 2001.
- [16]. Cover T. „Geometrical and statistical properties of systems of linear inequalities with applications in pattern recoquition”. IEEE Trans. Electronic Computers, 1965.

- [17]. Crawford G., Sen B., „Instrumenty pochodne. Narzędzie podejmowania decyzji finansowych”, Wydawnictwo K. E. Liber, 1998.
- [18]. Cristea A.I., Okamoto T., „NN for stock exchange prediction; a Lyapunov-based training” – Int. Conf. Computational Intelligence and Multimedia Applications, World Scientific, pp.416-21, ICCIMA, 1998.
- [19]. Deboeck G.: „Investment Maps of Emerging Markets”, in Deboeck G. and Kohonen T.: „Visual Explorations in Finance”, Springer 1997, pp. 83-105, 1997.
- [20]. Einstein A., „On the Movement of Small Particles Suspended in a Stationary Liquid Demanded by the Molecular-Kinetic Theory of Heat”, Ann. Physik 17, 1905.
- [21]. Elliot D.F., Rao K.R., „Fast transforms: algorithms, analysis, applications”, Acad. Press , 1982.
- [22]. Elton, Edwin J., Gruber, Martin J., „Nowoczesna teoria portfelowa i analiza papierów wartościowych”, WIG-PRESS, Warszawa 1998.
- [23]. Fausett L., „Fundamentals of Neural Networks: Architectures, Algorithms, and Applications”, Prentice Hall, 1994.
- [24]. Filipovic D., „Consistency Problems for Heath-Jarrow-Morton Interest Rate Models”, Springer-Verlag, Berlin – Heidelberg, 2001.
- [25]. Fisz M. „Rachunek prawdopodobieństwa i statystyka matematyczna”, PWN, Warszawa 1967.
- [26]. Gardiner C.W., „Handbook of Stochastic Methods: for Physics, Chemistry and the Natural Science”, Springer, 1996.
- [27]. Gardner E.S., „Exponential smoothing”, The state of the art. Journal if Forecasting, 4, 1-28, 1985.
- [28]. Goldberg D.E., „Algorytmy genetyczne i ich zastosowania” WNT, Warszawa, 1995.
- [29]. Haykin S. „Neural Networks, a Comprehensive Foundation” Macmillan College Publishing Company, New York 1994.
- [30]. Hebb D. „Organization of Behaviour”, New York 1949.
- [31]. Hertz J., Krogh A., Palmer R., „Wstęp do teorii obliczeń neuronowych”, WNT, Warszawa 1993.
- [32]. Hoff J.C., „A practical guide to Box-Jenkins forecasting”, Lifetime Learning Publications, London, 1983.

- [33]. Hołyst J.A., Hagel T., Haag H., Weidlich W., „How to Control Chaotic Economy?”, *J. Evolut. Econ.* 6, 31, 1996.
- [34]. Hołyst J.A., Hagel T., Haag G., „Destructive Role of Competition and Noise for Control of Microeconomical Chaos”, *Chaos, Solitons and Fractals* 8, 1489, 1997.
- [35]. Hornik K., Stinchcombe M., White H. „Multilayer feedforward networks are universal approximators”. *Neural Networks*, 1989.
- [36]. Hull J.C., „Options, Futures and Other Derivatives”, New Jersey, Prentice-Hall, 1993.
- [37]. Hull J.C., „Futures Options and Other Derivative Securities”, Prentice Hall, 1997.
- [38]. Hull J.C., „Kontrakty terminowe I opcje: wprowadzenie”, WIG-Press Warszawa, 1988.
- [39]. Hush D., Horne B., „Progress in supervised neural networks. *IEEE Signal Processing Magazine*”, 1993.
- [40]. Jajuga K., Jajuga T., „Inwestycje”, PWN Warszawa, 2000.
- [41]. Jakubowski J., Palczewski A., Rutkowski M., Stettner Ł., „Matematyka finansowa. Instrumenty pochodne”, WNT, Warszawa, 2003.
- [42]. Janicki A., Izydorczyk A., „Komputerowe metody w modelowaniu stochastycznym”, WNT, Warszawa, 2001.
- [43]. Jenkins, Watts, „Spectral Analysis and Its Applications”, Holden-Day, 1968.
- [44]. Jermakow S.M. „Metoda Monte Carlo i zagadnienia pokrewne”, PWN Warszawa, 1976.
- [45]. Kohonen T., „Self-organizing maps”. Springer-Verlag, Berlin, 1995.
- [46]. Lambert D.R., „Commodity Channel Index: Tool for Trading Cyclic Trends”, *Stocks and Commodities (Future Magazine)*, V.1:5(120-122), 1980.
- [47]. Lamberton D., Lapeyre B., „Introduction to Stochastic Calculus Applied to Finance”, Chapman & Hall, 1996.
- [48]. Leonard J.A., Kramer M.A. „Radial basis function networks for classifying process faults”. *IEEE Control System Magazine*, 1991.
- [49]. Lorenz H.W., „Nonlinear Dynamical Equations and Chaotic Economy”, Springer, 1993.
- [50]. Lula P., Morajda J., „Klasyfikacja wzorców występujących w finansowych szeregach czasowych przy użyciu sieci neuronowych Kohonena”, *Zeszyty naukowe AE w Krakowie nr 604, Prace z zakresu informatyki i jej zastosowań*, Kraków, 2002.

- [51]. Lula P., Tadeusiewicz R., „STATISTICA Neural Networks PL. Przewodnik problemowy”, StatSoft, Kraków 2001.
- [52]. Makridakis S., Wheelwright S.C., McGee V.E., „Forecasting: Methods and Applications (Wiley Series in Management)”, 1983.
- [53]. B.B. Mandelbrot, „Renormalization and Fixed Points in Finance, since 1962”, *Physica A*263, 477, 1999.
- [54]. McCulloch W.S., Pitts W., „A logical calculus of the ideas immanent in nervous activity”, *Bulletin of Mathematica Biophysics*, No 5, pp. 115-133, 1943.
- [55]. Minsky M., Papert S., „Perceptrons”, MIT Press, Cambridge 1969.
- [56]. Montgomery D.C., Johnson L.A., Gardiner J.S., „Forecasting & Time Series Analysis”, McGraw-Hill, New York, 1990.
- [57]. Morajda J., „Applications of neural networks in the financial markets - selected aspects” - Proceedings of the 4th Conference „Neural Networks and Their Applications” in Zakopane 18-22.05.1999, Częstochowa 1999.
- [58]. Morajda J., „Neural networks as predictive models in financial futures trading” - Proceedings of the 5th Conference „Neural Networks and Soft Computing” in Zakopane 6-10.06.2000, Częstochowa 2000.
- [59]. Morajda J., „Neural Networks and Their Economic Applications” – w: „Artificial Intelligence and Security in Computing Systems” (eds: J. Sołdek, L. Drobiazgowicz), Kluwer Academic Publishers, Boston/Dordrecht/London, 2003.
- [60]. Musiela M., Rutkowski M., „Martingale Methods in Financial Modelling”, Springer-Verlag, Berlin, 1998.
- [61]. Muth J.F., „Optimal Properties of Exponentially Weighted Forecasts”, *JASA*, 1960.
- [62]. Neumann J., „The Computer and the Brain”, Yale Univ. Press, New Haven, 1958.
- [63]. Ott E., Grebogi C., Yorke J.A., „Controlling chaos”, *Phys. Rev. Lett.* 64, 1196 1990.
- [64]. Pankratz A., „Forecasting with univariate Box-Jenkins models: Concepts and cases”, Wiley, New York, 1983.
- [65]. Peters E. „Teoria chaosu a rynki finansowe. Nowe spojrzenie na cykle, ceny i ryzyko”, WIG-Press, Warszawa 1997.
- [66]. Priestley M., „Spectral Analysis and Time Series: Univariate Series”, vol. 1, Academic, San Diego, 1981.

- [67]. Refenes A.P., Zaidi A., „Managing Exchange – Rate Prediction Strategies with Neural Networks. – Neural Networks Capital Markets”, Wiley, 1995.
- [68]. Rosenblatt F., „The perceptron. A theory of statistical separability in cognitive system”, Cornell Aeronautical Lab. Inc. Rep. No. VG-1196-G-1, 1965.
- [69]. Shumway R.H., „Applied statistical time series analysis”, Englewood Cliffs: Prentice-Hall, 1988.
- [70]. Smoluchowski M., „Zarys kinetycznej teorii ruchów Browna i roztworów mętnych”, Rozprawy wydziału matematyczno-przyrodniczego Akademii Umiejętności w Krakowie T XLVI, seria A, 257-281, 1906.
- [71]. Sobczyk K. „Stochastyczne równania różniczkowe”, PWN Warszawa, 1996.
- [72]. Sornette D., Johansen A., „A Hierarchical Model of Financial Crashes”, Physica A261, 581, 1998.
- [73]. Tarassenko L., Roberts S. „Supervised and unsupervised learning in radial basis function classifiers”, IEEE Proc. Vis. Image Signal Process, 1994.
- [74]. Taylor W.K., „Computers and the nervous system. Models and analogues in biology”, Cambridge Univ. Press, Cambridge, 1960.
- [75]. Trippi R., Turban E., „Neural Networks in Finance and Investing”, Probus Publishing Company, 1993.
- [76]. Vandaele W., „Applied Time Series and Box-Jenkins Models”, Academic Press Inc., 1983.
- [77]. Vapnik V.N., Chervonenkis A. „On the uniform convergence of relative frequencies of events to their probabilities. Theory of Probability and its Applications”, 1971.
- [78]. Velleman P. W., Hoaglin D.C., „Applications, Basics and Computing of Exploratory Data Analysis”, Boston, Duxbury Press, 1981.
- [79]. Welles, Wilder J., „New Concepts in Technical Trading Systems”, 1978.
- [80]. Weron A., Weron R., „Inżynieria finansowa”, WNT, Warszawa, 1999.
- [81]. Widrow B., Hoff M. „Adaptive switching circuits”, 1960.
- [82]. Wiener N., „Differential Space”, J. Math. Phys. 2, 131-174, 1923.
- [83]. Wierzbicki, M., „Analiza portfelowa”, MOTTE, Łódź 1995.
- [84]. Wilmott P., „Derivatives: The Theory and Practice of Financial Engineering”, John Wiley & Sons, Chichester, 1998.
- [85]. Zeliaś A., „Teoria prognozy”, PWE, Warszawa 1997.