

# Prognozowanie indeksu WIG20 za pomocą sieci neuronowych NARX i metody SVM

Sylwia Radomska\*

Nadesłany: 14 marca 2020 r. Zaakceptowany: 10 maja 2021 r.

---

## Streszczenie

Celem artykułu jest porównanie błędu prognoz indeksu WIG20 uzyskanych za pomocą tradycyjnej metody statystycznej oraz metod uczenia maszynowego: metody wektorów nośnych i metamodelu opartej na sieciach neuronowych NARX. Przeprowadzona analiza wskazuje, że metoda SVM pozwoliła na uzyskanie prognoz o największej precyzji (najniższych wartościach błędów *ex post*). Obie metody uczenia maszynowego cechowały się istotnie większą dokładnością prognoz w porównaniu z zastosowaną metodą statystyczną w okresie styczeń 2017 – marzec 2018 r. Tradycyjne modele statystyczne wymagają szeregu założeń dotyczących zależności pomiędzy zmiennymi, szeroko krytykowanych w literaturze za arbitralność. Charakterystyka modeli uczenia maszynowego z jednej strony podkreśla ich zdolność do wykrywania złożonych i nieliniowych zależności w danych historycznych, ale z drugiej strony wskazuje na ich inne ograniczenia metodologiczne.

---

**Słowa kluczowe:** prognozowanie, WIG20, sieci neuronowe (NARX), metoda wektorów nośnych (SVM), efektywność rynku

**JEL:** G17, G14, C45, C53

## 1. Wprowadzenie

Prognozowanie szeregów finansowych tradycyjnie oznacza w praktyce wykorzystanie tzw. analizy fundamentalnej oraz technicznej do przewidywania przyszłych wartości cen aktywów (Vanstone, Finnie 2009). Ujęcie tradycyjne to estymowanie za pomocą procesów stochastycznych parametrów szeregów finansowych wraz z oszacowaniami ich błędów losowych. Istotnym aspektem wykorzystywania prognoz do decyzji o transakcjach finansowych jest precyzja oszacowań. Prognoza ze (względnie) szerokim przedziałem ufności może oferować dodatnią wartość oczekiwaną z poziomem ryzyka prognozy powyżej preferencji graczy rynkowych względem ryzyka. Stąd zainteresowanie nie tylko samą wartością prognozowanych cen aktywów, lecz także niepewnością związaną z tą prognozą.

W ujęciu tradycyjnym, nawet gdy sposób estymacji uwzględnia nieliniowość, niestacjonarność oraz dynamiczną zmienność zależności między zmiennymi w modelu (Hsu i in. 2009), szacowane są zdefiniowane przez badacza *ex ante* parametry modelu i ich wariancja (na podstawie oszacowań wariancji składnika losowego modelu). Tymczasem dynamika procesów rynkowych oraz niezaniechana rola czynników politycznych czy katastrof naturalnych czynią zjawiska na rynkach finansowych potencjalnie nieliniowymi ze względu na parametry (Kumar, Murugan 2013). Ponadto nie wszystkie szeregi czasowe da się przedstawić za pomocą procesu stochastycznego. Wreszcie nawet jeśli odpowiednia formuła istnieje, jej „dokładne odkrycie” może być zbyt czasochłonne, by taki wysiłek miał praktyczną wartość.

Przewagą metod uczenia maszynowego nad metodami tradycyjnymi jest to, że pozwalają odtworzyć złożone, nieliniowe relacje pomiędzy zmiennymi na podstawie danych historycznych bez założenia, iż historyczne dane pochodzą z jednego i możliwego do zdefiniowania procesu stochastycznego (Atsalakis, Valavanis 2009). Spośród metod uczenia maszynowego do prognozowania szeregów finansowych najczęściej wykorzystywane są: metoda wektorów nośnych (SVM, por. m.in. Tay, Cao 2001; Kim 2003) oraz sieci neuronowe (Krollner, Vanstone, Finnie 2010). Metody te pozbawione są parametrycznych założeń dotyczących procesu stochastycznego generującego dane (Lu, Lee, Chiu 2009). Przykładowo, model nieliniowej autoregresyjnej sieci neuronowej (NARX – Nonlinear Autoregressive with Exogenous Input) pozwala osiągnąć prognozy o wysokiej precyzji nawet dla tzw. chaotycznych szeregów czasowych<sup>1</sup>. Z kolei metoda SVM jest bardzo specyficznym rodzajem algorytmów uczących się, który w przeciwieństwie do sztucznych sieci neuronowych charakteryzuje się rzadkością rozwiązania i wysoką odpornością na problem nadmiernego dopasowania (ang. *overfitting*).

Celem artykułu jest porównanie precyzji prognoz indeksu WIG20 uzyskanych za pomocą tradycyjnej metody statystycznej oraz dwóch metod uczenia maszynowego: sieci neuronowej oraz metody wektorów nośnych. Precyzję prognoz otrzymanych za pomocą wybranych metod porównano, wykorzystując miary *ex post*: m.in. średnią kwadratową sumę błędów oraz średni błąd bezwzględny. Wyniki wskazują, że model parametryczny cechuje wyższa wartość błędów *ex post* niż model SVM oraz model oparty na sieci NARX.

Artykuł ma następującą konstrukcję. W pierwszej części zostały omówione najważniejsze badania dotyczące prognozowania finansowych szeregów czasowych za pomocą metod uczenia maszynowego. W dalszej kolejności przedstawiono zastosowane modele uczenia maszynowego, najważniejsze informacje dotyczące szeregu finansowego WIG20 oraz dobór zmiennych wsadowych. Wszystkie modele

<sup>1</sup> Por. m.in. Diaconescu (2008). Przykładem są np. badania zmiennej przepływności ruchu wideo (Ardalani-Farsa, Zolfaghari 2010) czy plam słonecznych (Menezes, Barreto 2008). O zastosowaniach sieci NARX do rynków finansowych piszą m.in. Chaudhuri i Ghosh (2016) oraz Abdulkadir i Yong (2014).

zostały stworzone w środowisku programistycznym Matlab. Charakterystyki zbudowanych modeli oraz otrzymane za ich pomocą prognozy zostały przedstawione w rozdziale 5. Wyniki i wnioski z niniejszego badania zamykają artykuł.

## 2. Przegląd literatury

Metody uczenia maszynowego (jak również głębokiego uczenia maszynowego; ang. *deep learning*) w ostatnich kilkudziesięciu latach są coraz częściej wykorzystywane do prognozowania m.in. cen akcji, indeksów, kursów walutowych, zmienności czy trendów dotyczących różnych szeregów finansowych, a ich wyniki są konfrontowane z klasycznymi technikami wykorzystywanymi do prognozowania szeregów czasowych (Sezer, Gudelek, Ozbayoglu 2020). Systematyczny przegląd literatury dotyczącej wykorzystania sztucznych sieci neuronowych w ekonomii finansowej przedstawili Li i Ma (2010). Sezer, Gudelek i Ozbayoglu (2020) zaprezentowali najważniejsze publikacje dotyczące prognozowania finansowych szeregów czasowych za pomocą metod głębokiego uczenia maszynowego. W naszym badaniu wykorzystaliśmy sieć neuronową NARX i metodę SVM do prognozowania indeksu WIG20, dlatego w dalszej części artykułu zaprezentujemy wybrane publikacje dotyczące prognozowania finansowych szeregów czasowych za pomocą metody wektorów nośnych i sztucznych sieci neuronowych, w szczególności wykorzystanej w artykule sieci NARX.

Najważniejsze badania dotyczące prognozowania szeregów finansowych za pomocą metody SVM wraz z trzema istotnymi wnioskami przedstawili Jaramillo, Velasquez i Franco (2017). Po pierwsze, metodę wektorów nośnych najczęściej wykorzystuje się do prognozowania indeksów finansowych o jeden okres wprzód (prognozy dzienne). Po drugie, znaczna część publikacji poświęcona jest doborowi odpowiednich parametrów dla metody SVM. Po trzecie, metoda wektorów nośnych pozwala na dokładniejsze prognozy w porównaniu z klasycznymi metodami prognozowania szeregów czasowych oraz sieci neuronowych.

Sztuczne sieci neuronowe są wykorzystywane w świecie finansowym już od lat 90. (por. m.in. Trippi, Turban 1992; Azoff 1994; Kaastra, Boyd 1996; Haykin 1999). Systematyczny przegląd literatury dotyczący prognozowania finansowych szeregów czasowych za pomocą sztucznych sieci neuronowych zaprezentował Tealab (2018) wraz z wyszczególnieniem typów sieci zastosowanych w badaniach. Sieć neuronowa NARX, stosowana z sukcesem do prognozowania chaotycznych szeregów czasowych już od kilkunastu lat (Diaconescu 2008; Ardalani-Farsa, Zolfaghari 2010), została wykorzystana w niewielu badaniach dotyczących prognozowania finansowych szeregów czasowych (Soman 2008; Alkhoshi, Belkasim 2018). Soman (2008) wykorzystał sieć NARX do prognozowania kursów walutowych, a Alkhoshi i Belkasim (2018) prognozowali indeks Dow Jones. Oba badania dotyczyły prognozowania jeden okres do przodu oraz wykorzystywały tylko historyczne wartości prognozowanych szeregów czasowych.

Istnieją dwie główne kategorie dotyczące prognozowania finansowych szeregów czasowych: analiza jednowymiarowa i analiza wieloczynnikowa. Wykorzystanie tylko historycznych cen analizowanego szeregu finansowego do prognozowania za pomocą SVM przedstawili m.in. Cao i Tay (2001)<sup>2</sup>, Tay i Cao (2001), Kim (2003)<sup>3</sup> oraz Okasha (2014). Ich badania pokazują, że metoda SVM pozwala na

<sup>2</sup> Cao i Tay (2001) do prognozowania S&P 500 wykorzystali przekształcony szereg czasowy (m.in. średnią ruchomą oraz techniczne wskaźniki oparte na historycznych wartościach indeksu).

<sup>3</sup> Kim (2003) do prognozowania KOSPI wykorzystał 12 zmiennych technicznych obliczonych na podstawie cen historycznych indeksu.

dokładniejsze prognozy (charakteryzuje się mniejszymi błędami NMSE, MAE, DS, CP i CD) w porównaniu ze sztucznymi sieciami neuronowymi (m.in. wielowarstwowym perceptronem) czy klasycznymi modelami prognozowania szeregów czasowych (m.in. ARIMA). Ponieważ na ceny indeksów finansowych mają wpływ różne wydarzenia polityczne, ogólne warunki ekonomiczne czy oczekiwania traderów i inwestorów, niektóre z wcześniejszych badań stosujących metodę SVM do prognozowania indeksów wykorzystują inne zmienne ekonomiczne do prognozowania wybranych szeregów finansowych za pomocą metody wektorów nośnych (Huang, Nakamori, Wang 2005)<sup>4</sup>.

Nasze badanie ma istotny wkład w literaturę z trzech powodów. Po pierwsze, zgodnie z naszą najlepszą wiedzą nasze badanie jest pierwszym, które stosuje sieć neuronową NARX, wykorzystującą, poza historycznymi wartościami indeksu WIG20, szeregi czasowe różnych zmiennych finansowych i ekonomicznych do prognozowania indeksu giełdowego. Po drugie, zaproponowane przez nas modele prognozują horyzont czasowy składający się z pięciu okresów (dni) w przód. Większość badań dotyczących prognozowania finansowych szeregów czasowych za pomocą modeli uczenia maszynowego stosuje tylko prognozy dla jednego okresu wprzód (zazwyczaj są to prognozy dzienne) (Sezer, Gudelek, Ozbayoglu 2020). Po trzecie, zgodnie z naszą wiedzą jest to pierwsze badanie wykorzystujące metody uczenia maszynowego do prognozowania wartości indeksu WIG20.

### 3. Modele statystyczne oraz modele uczenia maszynowego

W artykule jako konwencjonalny model statystyczny wykorzystano klasyczny model regresji liniowej z dwóch powodów. Po pierwsze, jest to jedna z najczęściej stosowanych metod parametrycznych oraz najczęstszy punkt porównawczy dla sieci neuronowych (por. Atsalakis, Valavanis 2009). Po drugie, w naszym badaniu oprócz historycznych wartości szeregu czasowego WIG20 wykorzystano również inne zmienne ekonomiczne (rozdział 5), tj. modele uczenia maszynowego oraz modele regresji liniowej oparte na tych samych ekonomicznych szeregach czasowych. Klasyczne modele ARIMA, GARCH czy modele zmienności stochastycznej z założenia wykorzystują do prognozowania tylko historyczne wartości analizowanego szeregu (Box i in. 2015; Niu i in. 2020). By porównanie metod konwencjonalnych i uczenia maszynowego było miarodajne, modele powinny bazować na tym samym zestawie zmiennych objaśniających, tj. poza historycznymi wartościami WIG20 korzystać także z innych dostępnych szeregów ekonomicznych<sup>5</sup>.

Niech  $\{x_i, y_i\}_{i=1, \dots, n}$  oznacza zbiór  $n$  par obserwacji. Celem analizy jest znalezienie takiej zależności między zmienną objaśnianą  $y$  a zmienną objaśniającą  $x$ , która spełnia równanie liniowe względem parametrów  $a$  i  $b$ :  $y = a + bx + \epsilon$ , gdzie  $a$  jest stałą,  $b$  jest nachyleniem krzywej, a  $\epsilon$  reprezentuje różnice między wartościami dopasowanymi i faktycznymi  $y$  w danej próbie. Relację, która najlepiej odzwierciedla zależność między zmienną  $x$  i zmienną  $y$ , można otrzymać, formułując problem optymalizacji estymatorów parametrów  $\hat{a}$  i  $\hat{b}$ , tj. zminimalizowanie sumy odchyłeń obserwacji od relacji dla wszystkich obserwacji, czyli znalezienie minimum funkcji  $L = \sum_{i=1}^n \epsilon^2 = \sum_{i=1}^n (y_i - a - bx_i)^2$ . W praktyce oznacza to zróżniczkowanie funkcji  $L$  względem wektora estymatorów parametrów  $\hat{a}$  i  $\hat{b}$ . Przy takiej konstrukcji modelu nie jest możliwe uchwycenie nieliniowych zależności ze względu na parametry.

<sup>4</sup> Do prognozowania indeksu WIG20 wykorzystaliśmy 14 zmiennych ekonomicznych (więcej informacji w rozdziale 5).

<sup>5</sup> Naturalnym rozszerzeniem niniejszego badania byłoby uwzględnienie modeli ARIMA i GARCH wzbogaconych o analizę wieloczynnikową (*multivariate* ARIMA oraz *multivariate* GARCH) oraz porównanie precyzji prognoz z tych modeli z prognozami uzyskanymi metodami uczenia maszynowego.

Modele uczenia maszynowego nie nakładają takich restrykcji. Inspiracją dla stworzenia sztucznych sieci neuronowych – najstarszego z typów modeli uczenia maszynowego – było zrozumienie biologicznych aspektów działania ludzkiego mózgu (Haykin 1999). Sztuczne sieci neuronowe to modele matematyczne, które mogą być wykorzystywane do przetwarzania danych, m.in. potrafią dokonać klasyfikacji, rozpoznać wzorce występujące w danych albo pozwalają podjąć próbę prognozowania. Osowski (2006) podkreśla, że bardzo duża liczba neuronów i powiązań między nimi sprawia, że pojedyncze błędy popełnione przez neurony giną w ogólnej masie.

### 3.1. Sieci neuronowe

Sieci neuronowe zaczęły być z sukcesem wykorzystywane do prognozowania szeregów finansowych w latach 90. zeszłego stulecia (Azoff 1994). Krollner, Vanstone i Finnie (2010) podkreślają, że najczęściej stosowaną metodą prognozowania szeregów finansowych, spośród wszystkich metod uczenia maszynowego, są sieci neuronowe. Dzięki nieparametrycznemu wychwytywaniu nieliniowości najpopularniejsze sieci neuronowe – perceptron wielowarstwowy (ang. *multilayer perceptron*) i sieć jednokierunkowa (ang. *feedforward network*) – pozwalały na otrzymanie prognoz o niższym błędzie prognozy niż standardowe metody, np. metoda regresji, ARMA, ARIMA, ARCH czy GARCH (por. m.in. Kohzadi i in. 1996; Pacelli 2012).

W ciągu ostatnich kilkunastu lat w literaturze zaczęto rozwijać modele hybrydowe, które łączą metody uczenia maszynowego i metody ekonometryczne. Pionierska praca obejmuje połączenie popularnej liniowej metody ARIMA z sieciami neuronowymi (Zhang 2003). Taka metoda sprawdza się w szczególności dla złożonych problemów, w których występuje kombinacja zarówno liniowych, jak i nieliniowych struktur. Prognozy otrzymane metodą hybrydową były dokładniejsze od otrzymanych oddzielnie za pomocą metody ARIMA oraz sieci neuronowych. Do prognozowania chaotycznych szeregów czasowych Ardalani-Farsa i Zolfaghari (2010) wykorzystali metodę hybrydową – sieć neuronową Elmana oraz nieliniowy model autoregresyjny NARX (Nonlinear Autoregressive with Exogenous Input). Ich analiza pokazała, że sieć neuronowa NARX-Elmana sprawdziła się jako narzędzie prognozytyczne zarówno w odniesieniu do sztucznie stworzonych chaotycznych szeregów, jak i rzeczywistych szeregów chaotycznych (np. szeregu plam słonecznych).

Do prognozowania wartości szeregu WIG20 w niniejszym artykule wykorzystano nieliniową autoregresyjną sieć neuronową (NARX), która jest rekurencyjna i dynamiczna. Ma ona sprzężenia zwrotne, które dotyczą kilku warstw sieci. W celu uzyskania pełnej wydajności sieci neuronowej NARX dla prognozy szeregów czasowych wykorzystuje się przeszłe wartości rzeczywistego szeregu czasowego  $y(t)$  i wartości prognozowane  $\hat{y}(t)$ . Równania odpowiadające poszczególnym typom sieci można zapisać w następujący sposób (Boussaada i in. 2018):

$$\hat{y}(t+1) = F \left( \begin{array}{c} y(t), y(t-1), \dots, y(t-n_y), x(t+1) \\ x(t), x(t-1), \dots, x(t-n_x) \end{array} \right) \quad 3.1$$

$$\hat{y}(t+1) = F \left( \begin{array}{c} \hat{y}(t), \hat{y}(t-1), \dots, \hat{y}(t-n_y), x(t+1) \\ x(t), x(t-1), \dots, x(t-n_x) \end{array} \right) \quad 3.2$$

gdzie:

$F(\cdot)$  – funkcja mapująca sieci neuronowej,

$\hat{y}(t+1)$  – wartość prognozowana  $y$  dla okresu  $t+1$  otrzymana za pomocą sieci NARX,

$y(t), y(t-1), \dots, y(t-n_y)$  – rzeczywiste wartości analizowanego szeregu czasowego,

$\hat{y}(t), \hat{y}(t-1), \dots, \hat{y}(t-n_y)$  – wyniki sieci NARX,

$x(t), x(t-1), \dots, x(t-n_x)$  – wartości wejściowe sieci NARX, gdzie  $n_x$  to liczba opóźnień szeregu wejściowego, a  $n_y$  to liczba opóźnień szeregu wyjściowego.

Funkcja mapująca  $F(\cdot)$  jest nieznaną i przybliżaną w trakcie procesu uczenia sieci (Boussaada i in. 2018). W modelu sieci neuronowej NARX wewnętrzna architektura, która wykonuje to przybliżenie, to perceptron wielowarstwowy – MLP (ang. Multi Layer Perceptron). MLP może nauczyć się dowolnego typu ciągłego mapowania. W perceptronie wielowarstwowym neurony zorganizowane są warstwami. Oprócz warstwy wejściowej i wyjściowej istnieje co najmniej jedna warstwa ukryta.

W prognozach wykorzystuje się najczęściej NARX o architekturze szeregowo-równoległej, która wyróżnia się dwiema istotnymi zaletami. Po pierwsze, użycie wartości rzeczywistych jako danych wejściowych sieci zwiększa precyzję otrzymanych wyników. Po drugie, architektura takiej sieci jest jednokierunkowa, co pozwala na stosowanie zwykłych algorytmów szkoleniowych dla perceptronu wielowarstwowego. Po trzecie, po fazie treningu sieć NARX-MPL konwertowana jest do sieci o architekturze równoległej, co jest korzystne przy tworzeniu prognozy obejmującej kilka okresów wprzód. Dla sieci neuronowej NARX-MPL o szeregowo-równoległej architekturze przyszłe wartości szeregu czasowego  $y(t+1)$  są prognozowane na podstawie obecnych i przeszłych wartości szeregu  $x(t)$  oraz rzeczywistych wartości szeregu  $y(t)$ . W przypadku sieci o równoległej architekturze predykcja jest obliczana na podstawie obecnych i przeszłych wartości szeregu  $x(t)$  oraz przeszłych prognoz wartości szeregu  $\hat{y}(t)$ .

Zanim sieć neuronowa zostanie wykorzystana do rozwiązania problemu, należy wyznaczyć parametry dotyczące charakterystyk sieci neuronowej, m.in. architekturę sieci (liczba warstw ukrytych, liczba neuronów), typ funkcji aktywacji czy algorytm uczenia sieci. Problem doboru sieci neuronowej, jej architektury oraz algorytmu uczenia do różnych typów problemów jest rozważany już od kilku dekad. Na początku lat 90. Angus (1991) przedstawił kryteria doboru najlepszej sieci neuronowej w przypadku klasyfikacji i regresji nieliniowej. Kilka lat później Kaastra i Boyd (1996) zaprezentowali 8-krokową procedurę tworzenia sieci neuronowej do prognozowania finansowych i ekonomicznych szeregów czasowych, w której przedstawili zasady przygotowania danych, budowy sieci neuronowej (liczba warstw ukrytych, liczba neuronów, funkcja aktywacji) oraz proces uczenia sieci. Wybór odpowiedniej architektury sieci zależy od złożoności analizowanego problemu oraz jego typu (prognozowanie, klasyfikacja, analiza obrazu, analiza dźwięku etc.).

Ważnym elementem sieci neuronowej jest również funkcja aktywacji, która pozwala na znalezienie złożonych, wielowymiarowych powiązań funkcyjnych między sygnałami wejściowymi a wyjściowymi. Gomes, Ludermit i Lima (2011) zastosowali 12 różnych funkcji aktywacji i 7 różnych liczb warstw ukrytych w celu prognozowania finansowych szeregów czasowych<sup>6</sup>. Ponadto wykorzystali 2 różne algorytmy uczenia sieci i sprawdzili wyniki uzyskane za pomocą różnych kombinacji wspomnianych wcześniej charakterystyk sieci neuronowych. Pokazali, że wybór funkcji aktywacji oraz liczby warstw ukrytych w wielu przypadkach powinny zależeć od wyboru algorytmu uczenia sieci. Przy tworzeniu sieci neu-

<sup>6</sup> Analizie zostało poddanych 12 różnych szeregów finansowych, z których każdy charakteryzował się nadzwyczajnym zachowaniem.

ronowej należy również uważać na problem przeuczenia sieci, a możliwe sposoby na uniknięcie tego problemu przedstawili Srivastava i in. (2014).

Poza wyżej wymienionymi cechami sieci neuronowych ważny jest też wybór analizowanego horyzontu czasowego oraz podział analizowanego zbioru na uczący, treningowy i testowy. Optymalna wielkość zbioru treningowego pozwala nie tylko na osiągnięcie lepszych predykcji, ale także na zmniejszenie złożoności obliczeniowej – kosztu czasowego oraz pamięciowego (Walczak 2001). Chong, Han i Park (2017) przeanalizowali najnowsze badania dotyczące prognozowania cen aktywów za pomocą sieci neuronowych, m.in. pod kątem horyzontu czasowego danych historycznych oraz podziału na zbiory – treningowy, walidacyjny i testowy. Zróżnicowanie wielkości zbioru danych wejściowych jest bardzo duże: najdłuższy analizowany horyzont czasowy obejmował 30 lat, a najkrótszy niewiele ponad 2 lata. Walczak (2001) pokazał, że dla sieci neuronowej, której zadaniem jest prognozowanie szeregu finansowego, minimalny zbiór treningowy powinien składać się z danych obejmujących 2-letni okres, a dla sieci neuronowych opartych na metodzie propagacji wstecznej zazwyczaj mniejszy zbiór danych treningowych pozwalał uzyskać lepsze predykcje.

Predykcje szeregów finansowych otrzymane za pomocą sieci neuronowych nie są stabilne i mogą się znacznie różnić w zależności od podziału analizowanego zbioru na uczący, treningowy i testowy. Jednym z możliwych sposobów rozwiązania tego problemu jest zastosowanie metamodelowania. W tym celu najczęściej oblicza się prognozę jako średnią wielu prognoz otrzymanych za pomocą sieci neuronowych (Yu, Wang, Lai 2009).

### 3.2. Metoda wektorów nośnych

Jedną z najbardziej rozwojowych, z sukcesem wykorzystywanych do prognozowania szeregów finansowych, jest metoda wektorów nośnych SVM (ang. *support vector machines*; por. Tay, Cao 2001). Prognozy otrzymane za pomocą metody SVM były znacznie lepsze w porównaniu z otrzymanymi za pomocą sieci neuronowej opartej na metodzie propagacji wstecznej (Kim 2003). Sheta, Ahmed i Faris (2015) porównali dokładność prognoz otrzymanych przy użyciu regresji, sieci neuronowej (perceptronu wielowarstwowego MLP) i metody wektorów nośnych. Ich badanie pokazało, że metoda SVM z gaussowską funkcją jądra (RBF) miała najmniejsze błędy prognoz dla indeksu S&P 500. Badanie przeprowadzone przez Huang, Nakamori i Wang (2005) pokazało, że model oparty na metodzie wektorów nośnych najlepiej przewidywał kierunek zmian indeksu NIKKEI 225 w porównaniu z liniową i kwadratową analizą dyskryminacyjną oraz siecią neuronową.

Ta metoda uczenia nadzorowanego, szczególnie skuteczna w problemach klasyfikacji oraz aproksymacji, polega na maksymalizowaniu marginesu separacji np. między dwiema zadanymi klasami. Aproksymacja w SVM sprowadza problem dopasowania do zadania analogicznego do klasyfikacji. Podstawową ideą SVM jest przeniesienie zbioru treningowego do przestrzeni o większym wymiarze; w tym celu wykorzystuje się funkcję mapującą. Uzasadnieniem matematycznym jest twierdzenie Covera (1965), które dowodzi, że wzorce nieseparowalne liniowo w danej przestrzeni można przetransformować w inną przestrzeń o tym samym bądź wyższym wymiarze, w której z dużym prawdopodobieństwem staną się separowalne liniowo. Następnie za pomocą regresji liniowej wykonywana jest separacja danych w przestrzeni o wyższym wymiarze. Mapowanie danych następuje przy wykorzystaniu określonej funkcji zwanej funkcją jądra (ang. *kernel function*). Jako funkcję jądra najczęściej stosuje się

funkcje wielomianowe, normalne (gaussowskie) i sigmoidalne. Separacja danych odbywa się poprzez znalezienie optymalnej hiperpłaszczyzny (tzw. wektora podtrzymującego, ang. *support vector*), która maksymalizuje margines separacji.

Sieci SVM mogą być wykorzystane do aproksymowania wartości danych pomiarowych  $(x_i, d_i)$ , gdzie  $x_i$  jest wektorem wejściowym, a  $d_i$  jest wartością zadaną przyjmującą dowolną wartość ze zbioru liczb rzeczywistych. Niech  $\mathbf{x}$  oznacza wektor wejściowy. Po rzutowaniu go na przestrzeń  $K$ -wymiarową otrzymujemy zbiór cech  $\varphi_j(\mathbf{x})$  dla  $j = 1, 2, \dots, K$ , które opisują wektor początkowy  $\mathbf{x}$ . Początkowe równanie hiperpłaszczyzny w przestrzeni liniowej, określone wzorem  $g(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b = 0$ , może zatem zostać zastąpione przez następujące równanie:

$$g(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + b = \sum_{j=1}^N w_j \varphi_j(\mathbf{x}) + b = 0 \quad (3.3)$$

gdzie  $w_j$  oznaczają wagi prowadzące od  $\varphi_j(\mathbf{x})$  do neuronu wyjściowego, a  $b$  oznacza wagę polaryzacji.

Celem metody wektorów nośnych jest znalezienie liniowej funkcji spełniającej nierówności:  $f(\mathbf{x}) \leq \mathbf{w}$  oraz  $\varphi_j(\mathbf{x}) > b$ . Parametry funkcji jąder oraz wektor wagowy  $\mathbf{w}$  dobrane są w taki sposób, aby zminimalizować wartość funkcji błędów:

$$R(C) = \frac{C}{n} \sum_{i=1}^p L_{\epsilon}(d_i, f(x_i)) + \frac{1}{2} \|\mathbf{w}\|^2 \quad (3.4)$$

gdzie  $L_{\epsilon}$  jest funkcją błędów z tolerancją  $\epsilon$  wprowadzoną, aby sformułować problem uczenia jako zadanie programowania liniowego i zwiększyć efektywność działania sieci, która jest zdefiniowana w następujący sposób:

$$L_{\epsilon}(d, f(\mathbf{x})) = \begin{cases} |d - f(\mathbf{x})| - \epsilon & \text{dla } |d - f(\mathbf{x})| \geq \epsilon \\ 0 & \text{dla } |d - f(\mathbf{x})| < \epsilon \end{cases} \quad (3.5)$$

W celu zmierzenia stopnia niedopasowania modelu do danych i osiągnięcia akceptowalnego poziomu błędów wprowadza się zmienne dopełniające  $\xi_i$  oraz  $\xi_i^*$ . Wprowadzenie dodatkowych zmiennych sprowadza problem optymalizacji do znalezienia minimum wartości funkcji  $R(\mathbf{w}, \xi_i^*) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*)$  przy ograniczeniach:

$$\begin{cases} y_i - \langle \mathbf{w}, x_i \rangle - b \leq \epsilon + \xi_i \\ \langle \mathbf{w}, x_i \rangle + b - y_i \leq \epsilon + \xi_i \\ \xi_i, \xi_i^* \geq 0 \end{cases} \quad (3.6)$$

gdzie  $C$  jest stałą regularyzacyjną większą od zera, która ustala kompromis pomiędzy wartością funkcji błędów  $R$  i wartością marginesu separacji. Zmienna  $C$  reprezentuje karę za błąd predykcji większy od  $\epsilon$ . Celem SVM jest minimalizacja  $\xi_i$  oraz  $\xi_i^*$  oraz  $\mathbf{w}$ . Powyższy problem optymalizacji można rozwiązać, używając metody mnożników Lagrange'a.



Metoda SVM ma wiele zalet w porównaniu z innymi metodami klasyfikacji opartymi na uczeniu maszynowym, takimi jak sieci neuronowe, drzewa decyzyjne i inne. Po pierwsze, jest wydajna obliczeniowo w przestrzeniach wielowymiarowych. Ponadto wektory pomocnicze wykorzystują mały podzbiór danych treningowych, które dają metodzie SVM dużą przewagę nad pozostałymi metodami uczenia maszynowego i modelami konwencjonalnymi. SVM wyróżnia się dobrą generalizacją, ponieważ wykazuje małą wrażliwość na liczbę danych uczących (Osowski 2006). Zaletą sieci SVM w porównaniu z MLP jest także fakt, że sprowadzają problem uczenia do programowania kwadratowego, co najczęściej gwarantuje ściśle jedno minimum funkcji celu oraz brak związku pomiędzy punktem startu i punktem końcowym. W przypadku sieci neuronowych MLP wybór punktu startowego ma istotny wpływ na jakość otrzymanego rozwiązania. Ponadto w SVM nie ma związku pomiędzy liczbą szacowanych parametrów a miarą złożoności problemu optymalizacyjnego.

#### 4. Warszawski Indeks Giełdowy WIG20

Analizie poddaliśmy szereg cen indeksu WIG20 publikowanego przez Giełdę Papierów Wartościowych w Warszawie. WIG20 jest kwotowany od 16 kwietnia 1994 r. Za bazową wartość przyjęto 1000 punktów. Skonstruowany został na wzór amerykańskiego S&P 500 czy brytyjskiego FTSE 100. WIG20 jest indeksem typu cenowego, czyli do obliczania wartości indeksu wykorzystuje się tylko ceny zawartych w nim transakcji i nie uwzględnia się dochodów z tytułu dywidend. WIG20 jest również indeksem typu *blue-chip*, tj. notuje 20 największych i najbardziej płynnych polskich spółek akcyjnych, przy czym najwyżej 5 spółek może pochodzić z tego samego sektora. W związku z powyższym do prognozowania wartości WIG20 jako zmienne objaśniające wykorzystano wybrane ceny akcji spółek wchodzących w skład indeksu (rozdział 5).

#### 5. Zastosowanie wybranych metod do prognozowania wartości cen WIG20

Do prognozowania wartości WIG20 zastosowano wskaźniki finansowe, ekonomiczne, ceny akcji spółek wchodzących w skład WIG20 oraz wartości głównych indeksów światowych. Wykorzystano ceny zamknięcia 14 zmiennych ekonomicznych: kurs EURPLN, kurs USDPLN, ceny złota, ceny ropy, rentowność 2-letnich obligacji polskich, wartości indeksu DAX, PX, NIKKEI225, S&P500, ceny spółek KGHM, PKN Orlen, PKO BP, PZU<sup>7</sup>, Pekao. Do zbudowania modeli wykorzystano dane z 5-letniego okresu (od 2 stycznia 2012 r. do 30 grudnia 2016 r.). Przeprowadzono prognozę dla ponadrocznego okresu od 2 stycznia 2017 r. do 28 marca 2018 r. Wykres 1 przedstawia wartości indeksu WIG20 dla analizowanego horyzontu czasowego, z wyszczególnieniem okresu, dla którego wykonano prognozy.

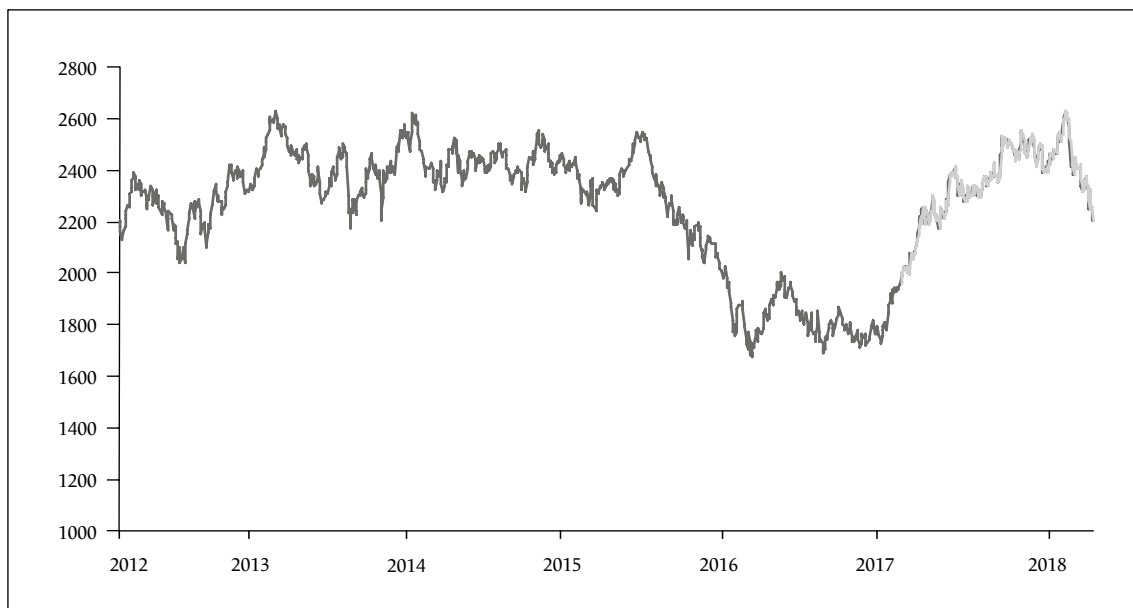
Analizowany ponadroczny okres podzielono na horyzonty 5-dniowe – wykonano prognozę okresów tygodniowych, stosując strategię bezpośrednią (ang. *direct strategy*), która polega na prognozowaniu każdego horyzontu niezależnie od pozostałych (por. m.in. Hamzaçebi, Akay, Kutay 2009). W tym celu zostało stworzonych 5 modeli dla metody regresji i 5 dla metody SVM oraz 500 sieci neuronowych NARX (dla każdego horyzontu po 100 sieci neuronowych), które prognozowały odpowiednio horyzont

<sup>7</sup> W przypadku spółki PZU nastąpił split akcji w październiku 2013 r. W celu ujednolicenia danych dokonano podziału akcji również dla wcześniejszego okresu.

1-dniowy, 2-dniowy, 3-dniowy, 4-dniowy oraz 5-dniowy. Po upływie każdego tygodnia wykorzystywano najnowsze dane do wykonywania kolejnych prognoz, nie zmieniając postaci funkcyjnej wszystkich zbudowanych modeli.

#### Wykres 1

Ceny indeksu WIG20 dla okresu od 2 stycznia 2012 do 29 marca 2018 r.



Uwaga: kolorem czarnym oznaczono dane wykorzystane do stworzenia modeli, a kolorem szarym wykorzystane do oszacowania błędów prognozy *ex post*.

Ważnym elementem projektowania sieci neuronowej jest dobór liczby neuronów w warstwie ukrytej oraz liczby opóźnień. Zbyt duża liczba neuronów może istotnie wydłużyć czas obliczeń oraz spowodować zbyt duże dopasowanie do danych (ang. *overfitting*). Z kolei zbyt mała liczba neuronów w warstwie ukrytej może uniemożliwić znalezienie odpowiednich zależności między danymi wejściowymi i wyjściowymi. Kaastra i Boyd (1996) zaproponowali kilka „reguł kciuka”, które można wykorzystać przy wyznaczaniu liczby neuronów ukrytych w każdej warstwie. Wilamowski (2009) pokazał, że mniejsza liczba neuronów w warstwach sieci wpływa pozytywnie na wyniki sieci – predykcje będą dokładniejsze i bardziej stabilne. W niniejszym artykule porównano dokładność prognoz otrzymanych za pomocą kilkunastu sieci NARX złożonych z różnej liczby neuronów w warstwach ukrytych. Na podstawie przeprowadzonej analizy wybrano sieci złożone z liczby neuronów pozwalających na uzyskanie najniższych wartości MSE (kryterium wyboru modelu: patrz tabela 2).

Predykcje uzyskane za pomocą sieci NARX były niestabilne i znacznie się różniły w zależności od podziału zbioru na uczący, treningowy i testowy. W związku z tym prognozę obliczono, wykorzystując metamodel oparty na sieci neuronowej NARX – jako średnią ze 100 prognoz otrzymanych za pomocą sieci neuronowych NARX (por. m.in. Yu, Wang, Lai 2009). Otrzymane modele regresji, NARX oraz SVM umożliwiają prognozę cen WIG20 na podstawie danych historycznych.

Wszystkie modele zostały stworzone w programie Matlab. Tabela 2 przedstawia informacje dotyczące charakterystyk stworzonych modeli. Modele regresji zakładają *explicite* liniowość ze względu na parametry. Model regresji został wybrany metodą krokową (ang. *stepwise*)<sup>8</sup>. Dla sieci neuronowych NARX jako funkcję aktywacji wykorzystano funkcję *symmetric hard-limit transfer function*, a dla metody SVM radialną funkcję bazową (RBF). W celu oceny precyzji otrzymanych wyników oraz porównania dokładności powyższych metod zostały wykorzystane kryteria porównawcze *ex post*. Kryteria są oparte na zmierzeniu bliskości przewidywanych wartości uzyskanych za pomocą określonego modelu w stosunku do rzeczywistych wartości cen WIG20. W tabeli 3 przedstawiono wyniki wybranych błędów *ex post* dla obu analizowanych okresów.

Tabela 1  
Charakterystyki stworzonych modeli

Cechy	Regresja	NARX	SVM
Charakterystyki stworzonych modeli	Regresja liniowa ( <i>stepwise</i> )	Algorytm uczenia: Levenberga-Marquardta Funkcja aktywacji: <i>hardlims</i> Liczba warstw ukrytych: 27 Liczba opóźnień: 1	Funkcja jądra: RBF
Kryterium wyboru modelu	RMSE	MSE	RMSE
Liczba stworzonych modeli dla każdego okresu	5	500	5
Podział zbioru na podzbiory	Tak, na 5 podzbiorów (w sposób losowy)	Tak, na 3 podzbiory 70%:15%:15% (w sposób losowy)	Tak, na 5 podzbiorów (w sposób losowy)
Dane	Ceny zamknięcia 14 zmiennych finansowych i ekonomicznych oraz indeksu WIG20 z dnia poprzedniego		
Okno estymacji	Rozszerzające		
Prognozy	5-dniowe		

W analizowanym okresie metoda wektorów nośnych pozwoliła na uzyskanie prognoz o największej precyzji – otrzymano najniższe wartości średniej kwadratowej błędów (RMSE), średniego błędu bezwzględnego (MAE), znormalizowanego błędu średniokwadratowego (NMSE) oraz średniego bezwzględnego błędu procentowego (MAPE). Niewiele wyższe wartości błędów otrzymano dla metamodelu opartego na sieciach neuronowych NARX. Najwyższą wartość błędów *ex post* otrzymano dla modelu parametrycznego.

Statystyka DS (ang. *directional symmetry*) jest miarą wydajności modelu w przewidywaniu kierunku zmian wartości rzeczywistego szeregu. Wyższe wartości wskazują, że znak zmiany wartości

<sup>8</sup> Metoda *stepwise* jest połączeniem metody *forward* i *backward*. W każdym kroku algorytmu jest dodawana nowa zmienna oraz usuwana jedna lub wiele zmiennych (dla których *p-value* jest większa niż kryterium wyjścia). Gdy do modelu nie można już dodać żadnej dodatkowej zmiennej (ze względu na kryterium wyjścia), algorytm jest zakończony.

z jednego okresu do następnego był częściej taki sam zarówno dla rzeczywistego, jak i przewidywanego szeregu czasowego. W analizowanym okresie model regresji pozwolił na uzyskanie najwyższej wartości miary symetrii kierunkowej. Ponadto dla wszystkich modeli statystyka DS pozostawała na poziomie poniżej 50, co oznacza, że więcej niż w połowie przypadków modele błędnie zaprognozowały zmianę ceny. Prognozowanie wartości indeksu oraz kierunku zmiany ceny to rozbieżne zadania prognostyczne, a najlepsze modele są wybierane na podstawie innych kryteriów (Huang, Nakamori, Wang 2005). Ponieważ celem stworzonych modeli było prognozowanie cen indeksu WIG20, a nie kierunku zmiany ceny, stworzone przez nas modele nie charakteryzują się wysokimi wartościami statystyki DS. Przeprowadzono również test (wartości testu *p-value* umieszczono w tabeli 3), który zweryfikował, że uzyskane prognozy kierunku zmian cen indeksu za pomocą wszystkich trzech metod nie różnią się istotnie od prognoz uzyskanych w sposób losowy, zakładających takie same prawdopodobieństwo spadku i wzrostu cen indeksu (gdy wartość statyki DS = 50). Hipoteza zerowa testu zakłada, że miara DS dla danej metody nie różniła się w istotny sposób od wartości 50. Otrzymane wyniki testu wskazują na brak podstaw do odrzucenia hipotezy zerowej.

Tabela 2

Porównanie błędów prognozy dla wybranych modeli dla okresu 2 stycznia 2017 – 28 marca 2018 r.

Model	Regresja	NARX	SVM
$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$	9271,60	3196,23	1507,27
$MAE = \frac{1}{n} \sum_{i=1}^n  (y_i - \hat{y}_i) $	323,14	164,92	70,16
$NMSE = \frac{1}{n} \sum_{i=1}^n \frac{ (y_i - \hat{y}_i) }{\left(\frac{1}{N} \sum_{i=1}^n y_i\right) \left(\frac{1}{N} \sum_{i=1}^n \hat{y}_i\right)}$	0,000069	0,000035	0,000015
$DS = \frac{100}{n-1} \sum_{i=2}^n d_i$	49,03 (0,41)*	48,06 (0,31)*	46,77 (0,21)*
$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left  \frac{y_i - \hat{y}_i}{y_i} \right $	13,36%	6,88%	2,92%

Uwagi:

$y$  oznacza dane, a  $\hat{y}$  prognozę;

$$d_i = \begin{cases} 1, & \text{gdy } (y_i - y_{i-1})(\hat{y}_i - \hat{y}_{i-1}) > 0 \\ 0, & \text{gdy } (y_i - y_{i-1})(\hat{y}_i - \hat{y}_{i-1}) < 0 \end{cases}$$

\* W nawiasach podano wartości *p-value* dla testu weryfikującego, czy wartości miary DS istotnie różnią się od wartości 50.

Prognozy uzyskane za pomocą sieci NARX, jak też metody wektorów nośnych charakteryzowały się niższymi wartościami błędów *ex post* (tabela 3) w porównaniu z metodą regresji w analizowanym okresie. W celu sprawdzenia, czy prognozy uzyskane analizowanymi metodami są od siebie statystycz-

nie różne, wykorzystano test Diebolda-Mariano (Diebold, Mariano 1995), który porównuje dokładność predykcji uzyskanych za pomocą dwóch metod. Na podstawie testu można stwierdzić, czy dwie prognozy szeregów czasowych charakteryzują się taką samą czy znacznie odmienną dokładnością predykcji.

Tabela 3

Wartości statystyk testu Diebolda-Mariano

Test Diebolda-Mariano		
Regresja vs. SVM	SVM vs. NARX	Regresja vs. NARX
2,93 (0,00)	5,14 (0,00)	2,73 (0,01)

Uwaga: w nawiasach podano wartości *p-value* dla testów istotności.

Hipoteza zerowa testu Diebolda-Mariano zakłada brak statystycznie istotnych różnic w dokładności prognoz otrzymanych za pomocą różnych metod. Przy poziomie istotności  $\alpha = 0,05$  należy odrzucić hipotezę zerową, jeżeli wartość statystyki  $DM > 0,84$ . Na podstawie wyników testu Diebolda-Mariano przedstawionych w tabeli 3 można stwierdzić, że dokładność prognoz otrzymanych za pomocą trzech analizowanych modeli była istotnie różna w analizowanym okresie. Oznacza to, że prognozy uzyskane za pomocą metody SVM charakteryzowały się istotnie większą precyzją oszacowań niż uzyskane za pomocą regresji oraz sieci NARX.

## 6. Podsumowanie

Prognozowanie przyszłych wartości cen akcji, indeksów giełdowych czy innych instrumentów finansowych jest przedmiotem badań inwestorów i ekonomistów od wielu dziesięcioleci. W tym celu wykorzystuje się analizę fundamentalną i techniczną, która na podstawie danych historycznych pozwala przewidzieć przyszłe wartości cen aktywów (Vanstone, Finnie 2009). Wykorzystanie tradycyjnych modeli statystycznych do prognozowania przyszłych wartości indeksu wymaga szeregu założeń dotyczących natury modelowanych procesów. Zastosowanie modeli uczenia maszynowego pozwala na wykrycie tych zależności bez konieczności formułowania założeń i ujawnia szereg nieliniowych relacji między zmiennymi na podstawie danych historycznych.

Celem artykułu było porównanie precyzji prognoz otrzymanych za pomocą modeli uczenia maszynowego oraz modelu regresji. Wykorzystano do tego miary *ex post*, m.in. średnią kwadratową sumę błędów, średni bezwzględny błąd procentowy czy miarę symetrii kierunkowej. W artykule wykorzystano metodę statystyczną (regresję) oraz dwie metody uczenia maszynowego (metodę SVM oraz metamodel oparty na sieci NARX) do prognozowania wartości cen indeksu WIG20. Do budowy modeli wykorzystano zmienne ekonomiczne obejmujące wskaźniki finansowe i ekonomiczne, ceny akcji wybranych spółek oraz wartości głównych indeksów światowych. Zastosowane metody uczenia maszynowego oparto na funkcjach nieliniowych (jako funkcję aktywacji dla sieci NARX wykorzystano *symmetric hard-limit transfer function*, jako funkcję jądra dla metody wektorów nośnych – funkcję RBF).

W analizowanym okresie (2 stycznia 2017 – 28 marca 2018 r.) największą precyzją prognoz (najmniejsze wartości błędów prognozy *ex post*) charakteryzowała się metoda wektorów nośnych, natomiast najwyższe wartości błędów prognozy *ex post* uzyskano dla metody regresji. Przeprowadzony test Diebolda-Mariano wskazuje, że prognozy uzyskane za pomocą metody SVM są istotnie różne w porównaniu z dwiema pozostałymi metodami. Nasze badanie potwierdza wcześniejsze wnioski – uzyskane m.in. przez Cao i Tay (2001), Kim (2003) czy Okasha (2014) – że prognozy o większej precyzji (niższych błędach *ex post*) dla finansowych szeregów czasowych można uzyskać, stosując metodę SVM niż sieci neuronowe NARX.

Wyniki przeprowadzonej analizy potwierdzają, że zgodnie z dotychczasową literaturą jedną z głównych przewag metod uczenia maszynowego nad standardowymi metodami statystycznymi jest możliwość dużego dopasowania do danych oraz możliwość odtworzenia złożonych, nieliniowych relacji pomiędzy zmiennymi na podstawie danych historycznych, dzięki zastosowaniu nieliniowych funkcji aktywacji dla sieci neuronowych i funkcji jądra dla SVM, w szczególności w okresie dużej zmienności cen indeksu. Dalsze badania powinny objąć zastosowanie innych funkcji aktywacji dla sieci neuronowych i funkcji jądra dla metody wektorów nośnych oraz porównanie precyzji otrzymanych prognoz, jak również wykorzystanie innych metod statystycznych jako metod porównawczych.

## Bibliografia

- Abdulkadir S.J., Yong S.P. (2014), Empirical analysis of parallel-NARX recurrent network for long-term chaotic financial forecasting, *2014 International Conference on Computer and Information Sciences (ICCOINS)*, IEEE.
- Alkhoshi E., Belkasim S. (2018), Stable stock market prediction using NARX algorithm, in: *Proceedings of the 2018 International Conference on Computing and Big Data*.
- Angus J.E. (1991), *Criteria for choosing the best neural network. Part 1*, NHRC-91-16, Naval Health Research Center.
- Ardalani-Farsa M., Zolfaghari S. (2010), Chaotic time series prediction with residual analysis method using hybrid Elman-NARX neural networks, *Neurocomputing*, 73(13–15), 2540–2553.
- Atsalakis G.S., Valavanis K.P. (2009), Surveying stock market forecasting techniques. Part II: Soft computing methods, *Expert Systems with Applications*, 36(3), 5932–5941.
- Azoff E.M. (1994), *Neural Network Time Series Forecasting of Financial Markets*, John Wiley & Sons.
- Boussaada Z., Curea O., Remaci A., Camblong H., Bellaaj N.M. (2018), A nonlinear autoregressive exogenous (narx) neural network model for the prediction of the daily direct solar radiation, *Energies*, 11(3), 620.
- Box G.E., Jenkins G.M., Reinsel G.C., Ljung G.M. (2015), *Time Series Analysis: Forecasting and Control*, John Wiley & Sons.
- Cao L., Tay F.E. (2001), Financial forecasting using support vector machines, *Neural Computing & Applications*, 10(2), 184–192.
- Chaudhuri T.D., Ghosh I. (2016), Artificial neural network and time series modeling based approach to forecasting the exchange rate in a multivariate framework, *Journal of Insurance and Financial Management*, 5(1), 92–123.

- Chong E., Han C., Park F.C. (2017), Deep learning networks for stock market analysis and prediction: methodology, data representations, and case studies, *Expert Systems with Applications*, 83, 187–205.
- Cover T.M. (1965), Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition, *IEEE Transactions on Electronic Computers*, 3, 326–334.
- Diaconescu E. (2008), The use of NARX neural networks to predict chaotic time series, *WSEAS Transactions on Computer Research*, 3(3), 182–191.
- Diebold F.X., Mariano R.S. (1995), Comparing predictive accuracy, *Journal of Business & Economic Statistics*, 13(3), 253–263.
- Gomes G.S.D.S., Ludermir T.B., Lima L.M. (2011), Comparison of new activation functions in neural network for forecasting financial time series, *Neural Computing and Applications*, 20(3), 417–439.
- Hamzaçebi C., Akay D., Kutay F. (2009), Comparison of direct and iterative artificial neural network forecast approaches in multi-periodic time series forecasting, *Expert Systems with Applications*, 36(2), 3839–3844.
- Haykin S. (1999), *Neural Networks: A Comprehensive Foundation*, Englewood Cliffs, Prentice-Hall.
- Hsu S.-H., Hsieh J., Chih T.-C., Hsu K.-C. (2009), A two-stage architecture for stock price forecasting by integrating self-organizing map and support vector regression, *Expert Systems with Applications*, 36(4), 7947–7951.
- Huang W., Nakamori Y., Wang S.Y. (2005), Forecasting stock market movement direction with support vector machine, *Computers & Operations Research*, 32(10), 2513–2522.
- Jaramillo J., Velasquez J.D., Franco C.J. (2017), Research in financial time series forecasting with SVM: contributions from literature, *IEEE Latin America Transactions*, 15(1), 145–153.
- Kaasra I., Boyd M. (1996), Designing a neural network for forecasting financial and economic time series, *Neurocomputing*, 10(3), 215–236.
- Kim K.J. (2003), Financial time series forecasting using support vector machines, *Neurocomputing*, 55(1–2), 307–319.
- Kohzadi N., Boyd M.S., Kermanshahi B., Kaasra I. (1996), A comparison of artificial neural network and time series models for forecasting commodity prices, *Neurocomputing*, 10(2), 169–181.
- Krollner B., Vanstone B.J., Finnie G.R. (2010), *Financial Time Series Forecasting with Machine Learning Techniques: A Survey*, ESANN.
- Kumar D., Murugan S. (2013), Performance analysis of Indian stock market index using neural network time series model, in: *International Conference on Pattern Recognition, Informatics and Mobile Engineering*, IEEE.
- Li Y., Ma W. (2010), Applications of artificial neural networks in financial economics: a survey, *2010 International Symposium on Computational Intelligence and Design*, 1, IEEE.
- Lu C.J., Lee T.S., Chiu C.C. (2009), Financial time series forecasting using independent component analysis and support vector regression, *Decision Support Systems*, 47(2), 115–125.
- Menezes Jr, J.M.P., Barreto G.A. (2008), Long-term time series prediction with the NARX network: an empirical evaluation, *Neurocomputing*, 71(16–18), 3335–3343.
- Niu T., Wang J., Lu H., Yang W., Du P. (2020), Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting, *Expert Systems with Applications*, 148, 113237.
- Okasha M.K. (2014), Using support vector machines in financial time series forecasting, *International Journal of Statistics and Applications*, 4(1), 28–39.

- Oowski S. (2006), *Sieci neuronowe do przetwarzania informacji*, Oficyna Wydawnicza Politechniki Warszawskiej.
- Pacelli V. (2012), Forecasting exchange rates: a comparative analysis, *International Journal of Business and Social Science*, 3(10), 145–156.
- Sezer O.B., Gudelek M.U., Ozbayoglu A.M. (2020), Financial time series forecasting with deep learning: a systematic literature review: 2005–2019, *Applied Soft Computing*, 90, 106181.
- Sheta A.F., Ahmed S.E.M., Faris H. (2015), A comparison between regression, artificial neural networks and support vector machines for predicting stock market index, *Soft Computing*, 7(8), 55–63.
- Soman P.C. (2008), *An Adaptive NARX Neural Network Approach for Financial Time Series Prediction*, Rutgers University.
- Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R. (2014), Dropout: a simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research*, 15(1), 1929–1958.
- Tay F.E., Cao L. (2001), Application of support vector machines in financial time series forecasting, *Omega*, 29(4), 309–317.
- Tealab A. (2018), Time series forecasting using artificial neural networks methodologies: a systematic review, *Future Computing and Informatics Journal*, 3(2), 334–340.
- Trippi R.R., Turban E. (1992), *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real World Performance*, McGraw-Hill.
- Vanstone B., Finnie G. (2009), An empirical methodology for developing stock market trading systems using artificial neural networks, *Expert Systems with Applications*, 36(3), 6668–6680.
- Walczak S. (2001), An empirical analysis of data requirements for financial forecasting with neural networks, *Journal of Management Information Systems*, 17(4), 203–222.
- Wilamowski B.M. (2009), Neural network architectures and learning algorithms, *IEEE Industrial Electronics Magazine*, 3(4), 56–63.
- Yu L., Wang S., Lai K.K. (2009), A neural-network-based nonlinear metamodeling approach to financial time series forecasting, *Applied Soft Computing*, 9(2), 563–574.
- Zhang G.P. (2003), Time series forecasting using a hybrid ARIMA and neural network model, *Neurocomputing*, 50, 159–175.

## Podziękowania

Autorka dziękuje Joannie Tyrowicz i anonimowym recenzentom za cenne uwagi. Błędy i uchybienia obciążają wyłącznie autorkę.