

UNIwersYTET ŁÓDZKI
WYDZIAŁ MATEMATYKI I INFORMATYKI
Kierunek Analiza Danych

Anna Stykowska

PROGNOZOWANIE OBCIĄŻEŃ SYSTEMU
ELEKTROENERGETYCZNEGO W RÓŻNYCH
HORYZONTACH CZASOWYCH

praca magisterska
napisana pod kierunkiem
dr Sebastiana Sakowskiego

Łódź 2022

Spis treści

Wstęp	2
0.1 Cel pracy	2
1 Pojęcia wstępne	4
1.1 Sieci neuronowe	4
1.2 Przetwarzanie sekwencji	6
1.2.1 Rekurencyjne Sieci Neuronowe RNN	6
1.2.2 Architektura sieci LSTM	7
1.2.3 Architektura sieci GRU	10
2 Budowa modeli	12
2.1 Przygotowanie danych	12
2.2 Eksploracja danych	13
2.2.1 Wykresy pudełkowe	16
2.2.2 Współczynnik zmienności	19
2.2.3 Autokorelacja	19
2.3 Metryki oceny jakości modeli	21
2.3.1 Modele naiwne	22
2.4 Modele ARMA oraz ARIMA	23
2.4.1 Dobór parametrów modelu	25
2.4.2 Kryteria informacyjne	30
2.5 Modele sieci LSTM i GRU	31
2.5.1 Parametry modelu	31
2.5.2 Optymalizacja Bayesowska	31
2.5.3 Wybór najlepszego modelu	32
3 Rezultaty	35
3.1 Wizualizacje predykcji poszczególnych modeli	35
3.1.1 Porównanie modeli	38
3.2 Porównanie dla różnych horyzontów czasowych	39
3.3 Horyzont czasowy: 7 dni	40
3.4 Horyzont czasowy: 14 dni	40
3.5 Horyzont czasowy: 30 dni	41
3.6 Horyzont czasowy: 60 dni	41
3.7 Horyzont czasowy: 90 dni	41

3.8	Wpływ zmienności szeregu na wyniki predykcji	41
3.9	Wpływ wybranej normalizacji na wyniki modelu	42
4	Podsumowanie	44
	Literatura	45

Wstęp

Tematem pracy jest predykcja obciążeń systemu elektroenergetycznego w różnych horyzontach czasowych. W tym celu zostaną porównane różne modele wykorzystujące sztuczną inteligencję oraz metody statystyczne utworzone z zamiarem prognozowania kolejnych wartości szeregu czasowego.

Prognozy zapotrzebowania na energię elektryczną mają kluczowe znaczenie dla zbilansowania systemu elektroenergetycznego. Utrzymanie równowagi pomiędzy zużyciem a produkcją energii elektrycznej jest warunkiem koniecznym dla jego poprawnego działania. Niedokładne prognozy mogłyby skutkować przerwami w dostawach lub nadmiernym wytwarzaniem prądu, co prowadziłoby do znacznych strat finansowych. Istotne jest posiadanie dokładnych predykcji zarówno dla krótkich i długich horyzontów czasowych. Jest to szczególnie ważne dla przedsiębiorstw energetycznych, dla których konieczne jest zaplanowanie przyszłych inwestycji produkcyjnych oraz negocjowanie kontraktów (na podstawie prognoz długoterminowych, powyżej 1 roku) oraz zminimalizowanie ryzyka przerw w dostawie i nadprodukcji energii (opierając się na prognozach krótkoterminowych - godzinowych, dziennych i tygodniowych). Jednocześnie jest to wsparcie dla energetycznych zakładów przemysłowych w zakresie ustalania terminów konserwacji urządzeń, gromadzenia rezerw paliwowych, a także importu i eksportu energii. Dodatkowo, nie możemy zakładać niezmienności sezonowości i trendu na przestrzeni lat, ze względu na coraz większy udział źródeł energii odnawialnej na rynku, pozyskiwanej między innymi z elektrowni wiatrowych, słonecznych fotowoltaicznych i geotermalnych. Odpowiedni dobór modeli dla prognoz krótko- i długoterminowych jest ściśle związany z minimalizacją ryzyka finansowego i ryzyka wystąpienia usterek w systemie energetycznym, co skutkuje lepszą wydajnością oraz satysfakcjonującymi wynikami finansowymi[1].

Badane dane zostały udostępnione na stronie Europejskiej Sieci Operatorów Elektroenergetycznych Systemów Przesyłowych <https://www.entsoe.eu/data/power-stats/>. Obejmują one historyczne dane obciążeń systemów energetycznych, mierzonych co godzinę i rozdzielonych na poszczególne państwa. W celu analizy posłużono się pomiarami dotyczącymi Polski w latach 2006 - 2015.

0.1 Cel pracy

Głównym celem pracy jest predykcja obciążeń systemu elektroenergetycznego w różnych horyzontach czasowych. Opracowano odpowiednie modele znajdujące zastosowanie w predykcji kolejnych wartości szeregów czasowych, a wyniki porównano przy

pomocy różnych metryk ewaluacji. Oprócz standardowych miar błędu wykorzystano w celu porównań również 2 modele naiwne. Modele z najlepszymi parametrami wybrano bazując na wynikach kryteriów informacyjnych oraz opisanych optymalizatorów. Sprawdzono również jak rozproszenie danych wpływa na jakość predykcji oraz czy można poprawić modele stosując inne podejście do standaryzacji danych.

Rozdział 1

Pojęcia wstępne

Definicja 1.1 (Szereg czasowy [2])

Szeregiem czasowym nazywamy uporządkowany zbiór obserwacji x_t , z których każda zarejestrowana jest w określonym czasie t . Dyskretny szereg czasowy charakteryzuje się rejestrowaniem obserwacji z określonym krokiem czasowym.

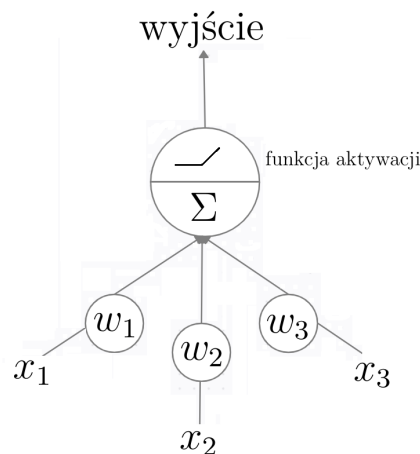
1.1 Sieci neuronowe

Pierwsze wzmianki związane z sieciami neuronowymi znajdują się w artykule "*A Logical Calculus of Ideas Immanent in Nervous Activity*"[3], napisanym przez Warrena McCullocha i Waltera Pittsa. Naukowcy opisali pracę neuronów w mózgu wykonującym złożone obliczenia logiczne z dziedziny rachunku zdań i zaproponowali ideę architektury sztucznych sieci neuronowych.

Definicja 1.2 (Perceptron)

Perceptron jest definiowany jako jednowarstwowa sieć neuronowa. Jest to jednoklasowy klasyfikator (binarny). Perceptrony wielowarstwowe nazywane są sieciami neuronowymi.

Neuron, jako podstawowa jednostka z której składają się perceptrony, został od tego czasu zmodyfikowany. Na wejściach i wyjściach przyjmowane są wartości liczbowe zamiast stanów binarnych, a każde połączenie w neuronie ma również własną wagę. Końcowy wynik uzyskujemy po zsumowaniu wejściowych sygnałów z uwzględnieniem wag i nałożeniu na tą sumę wybranej funkcji aktywacji[4, 5].



Rysunek 1.1: Schemat sztucznego neuronu

Na schemacie przyjęto następujące oznaczenia:

- x_1, x_2, x_3 - wartości wejściowe,
- w_1, w_2, w_3 - wagi przyporządkowywane odpowiednim wartościom,
- Σ - suma ważona

Perceptron składa się z jednej lub wielu (perceptron wielowarstwowy) warstw takich jednostek. Wszystkie wejścia w perceptronie przekazywane są do każdego neuronu - takie połączenie określane jest warstwą gęstą. Sygnały wejściowe są przekazywane do warstwy wejściowej i kolejno do warstw ukrytych (jeśli istnieją) i warstwy wyjściowej, w których wykonywane są obliczenia. Z uwagi na kierunek płynięcia sygnału od wejścia do wyjścia, sieć taką nazywamy jednokierunkową siecią neuronową. Do skutecznego uczenia perceptronów wielowarstwowych potrzebny jest jednak algorytm propagacji wstecznej, który jest w stanie zmniejszyć wartość uzyskiwanych błędów, modyfikując wagi połączeń[4, 5].

Definicja 1.3 (Propagacja wsteczna)

Propagacja wsteczna jest algorytmem uczenia wielowarstwowych sieci neuronowych. Dzięki niej ustalany jest właściwy kierunek zmiany wag w poprzednich warstwach sieci, który prowadzi do zminimalizowania otrzymanego błędu. Uzyskuje się to poprzez zmianę kierunku przepływu wektora informacji i adaptacji wag na podstawie przekazanej różnicy pomiędzy wartością predykowaną i właściwą oraz pochodnych obliczonych dla funkcji aktywacji danych warstw[6].

Pomimo zastosowania algorytmu propagacji wstecznej i skuteczności perceptronów wielowarstwowych w radzeniu sobie w zadaniach regresji oraz klasyfikacji, aby przetwarzać sekwencje i zauważać zależności pojawiające się w szeregach czasowych, zaistniała potrzeba utworzenia bardziej zaawansowanych - rekurencyjnych - sieci neuronowych.

1.2 Przetwarzanie sekwencji

1.2.1 Rekurencyjne Sieci Neuronowe RNN

RNN (ang. *recurrent neural network*) jest jedną z podstawowych architektur sieci neuronowych stosowanych do przetwarzania sekwencji danych i prognozowania szeregów czasowych. W przeciwieństwie do tradycyjnych sieci neuronowych, sekwencje wejściowe i wyjściowe nie są od siebie niezależne - sieć RNN zapamiętuje również informacje uzyskane z poprzednich kroków czasowych. Każdy neuron rekurencyjny w danym kroku czasowym otrzymuje do przetworzenia zarówno wyjście z kroku poprzedniego y_{t-1} jak i wejściowy wektor x_t . Odbywa się to za pomocą przyporządkowanym im wag. Jeśli macierze wektorów wag dla warstwy rekurencyjnej oznaczone zostaną jako W_x oraz W_y , wynik obliczeń można zapisać jako:

$$y_t = \phi(W_x^T x_t + W_y^T y_{t-1} + b), \quad (1.1)$$

gdzie:

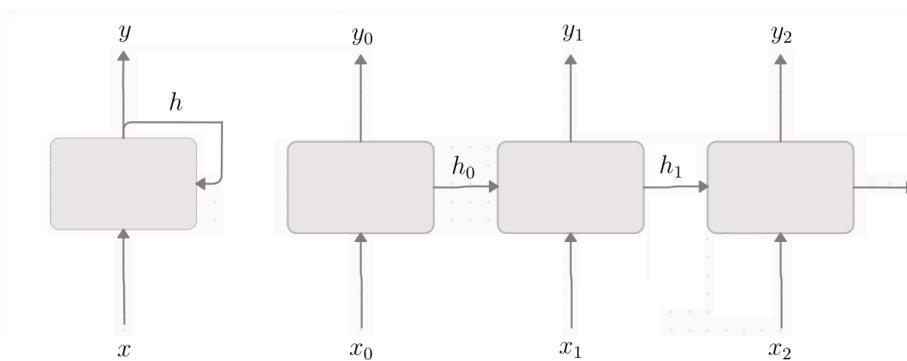
- ϕ - funkcja aktywacji,
- b - wektor obciążeń neuronów.

Definicja 1.4 (Komórka pamięci (ang. *memory cell*))

Jeden z podstawowych komponentów rekurencyjnych sieci neuronowych, który na pewnych przedziałach czasu zapamiętuje pewien stan.

Oznaczmy stan komórki w danym takcie jako h_t . Jest to połączenie danych z taktu poprzedniego oraz danych wejściowych z aktualnego kroku czasowego:

$$h_t = f(h_{t-1}, x_t). \quad (1.2)$$



Rysunek 1.2: Schemat działania podstawowej komórki RNN

Trenowanie rekurencyjnych sieci neuronowych odbywa się poprzez zastosowanie tzw. propagacji wstecznej w czasie - sieć jest najpierw rozwijana w czasie, a następnie stosowana jest propagacja wsteczna. Pierwszym krokiem jest przejście w przód po

rozwinętej sieci RNN i ocena wektora wyjściowego poprzez zastosowanie wybranej funkcji kosztu. W kolejnym kroku gradienty funkcji kosztu są rozprowadzane wstecznie po rozwinętej sieci, dzięki czemu parametry modelu mogą być zaktualizowane.

Problem niestabilnych gradientów

Kierunek i wielkość stosowaną podczas uczenia do poprawy wag sieci nazywa się gradientem błędu. Gdy rozpatrujemy głęboką rekurencyjną sieć neuronową, nietrudno spotkać się z problemem eksplodującego gradientu. Dzieje się to, gdy gradienty błędów się kumulują i powodują bardzo duże aktualizacje wag modelu sieci neuronowej podczas uczenia - otrzymujemy w ten sposób niestabilną sieć - wagi mogą osiągać nawet wartości NaN. NaN (ang. *not a number*) jest numerycznym typem danych, który używany jest do reprezentacji dowolnej wartości, która jest niezdefiniowana lub niemożliwa do przedstawienia - w tym wypadku są to zbyt duże wartości. W takiej sytuacji waga nie może być już zaktualizowana. Istnieją różne metody radzenia sobie z eksplodującymi gradientami, takie jak normalizacja warstw lub obcinanie gradientów.

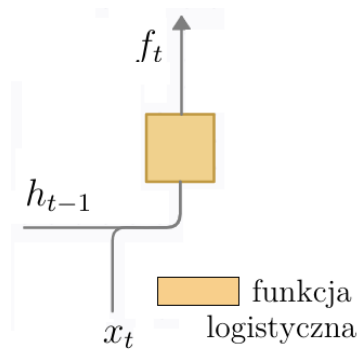
Pamięć krótkotrwała

Drugim problemem sieci RNN podczas przetwarzania sekwencji jest pamięć krótkotrwała. Podczas stopniowego przepływu danych przez zbudowaną sieć, model stopniowo traci najstarsze informacje, a po upływie odpowiednio dużej liczby kroków, całkiem zapomina te początkowe. W celu poradzenia sobie z tym zagadnieniem, zostały zaprojektowane specjalne komórki pamięci długotrwałej, do których należą między innymi komórki LSTM oraz GRU opisane w dalszej części pracy[7].

1.2.2 Architektura sieci LSTM

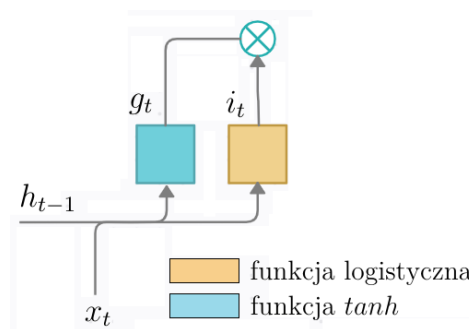
Sieci LSTM (ang. *Long - short term memory*) to specjalny rodzaj rekurencyjnej sieci neuronowej, zaproponowany i opisany przez Hochreitera i Schmidhubera [8]. Architektura sieci umożliwia naukę zależności zarówno w krótkich, jak i w długich okresach czasu, z czym tradycyjna sieć RNN radzi sobie gorzej. Efektywność i zoptymalizowanie czasu obliczeń uzyskuje dzięki wprowadzeniu bramek, pełniących określone funkcje: [9]

- bramka zapominająca - jest odpowiedzialna za usuwanie informacji ze stanu komórki. Po wykonaniu odpowiednich obliczeń szacuje, które informacje są mało ważne lub niepotrzebne LSTM do dalszej nauki. Pomaga to osiągnąć lepszą wydajność sieci.



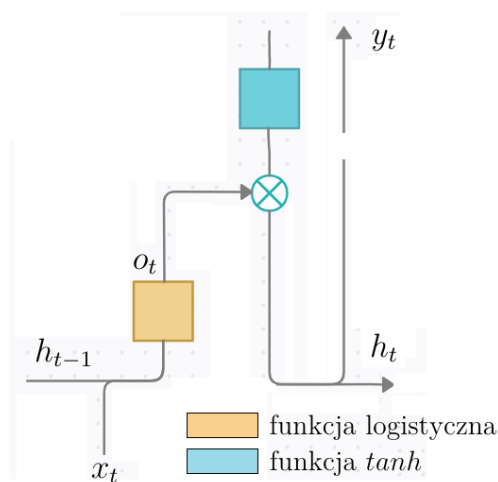
Rysunek 1.3: Schemat bramki zapominającej

- bramka wejściowa - odpowiada za wybranie tych informacji, które powinny zostać dołączone do stanu długotrwałego.



Rysunek 1.4: Schemat bramki wejściowej

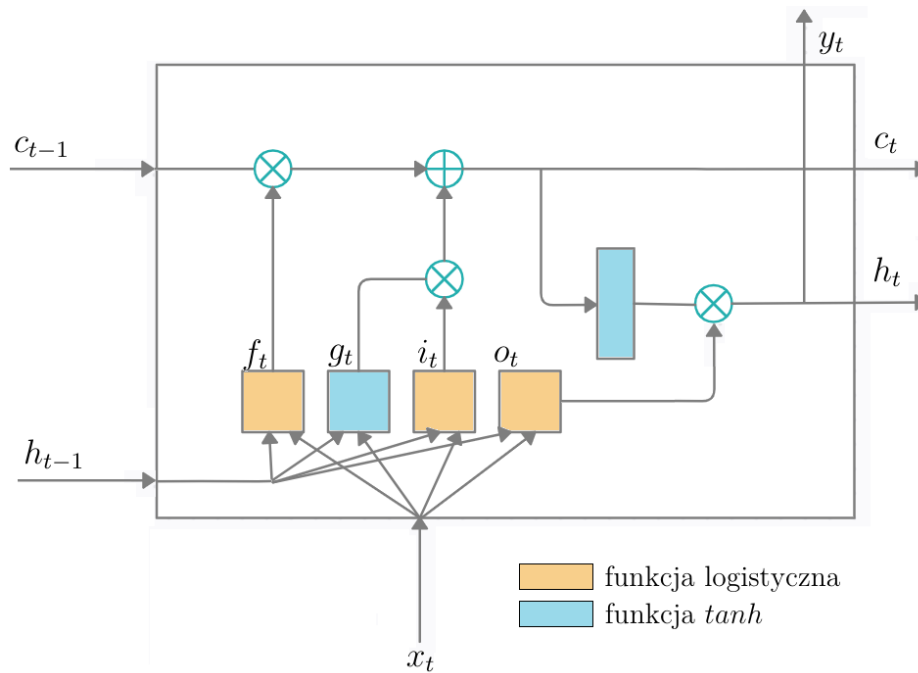
- bramka wyjściowa - określa pod koniec każdego taktu, które części stanu długotrwałego powinny zostać przekazane do następnego kroku.



Rysunek 1.5: Schemat bramki wyjściowej

Na schematach zastosowane zostały następujące obliczenia:

- h_t - wektor stanu krótkotrwałego
- c_t - wektor stanu długotrwałego
- x_t - wektor danych wejściowych
- y_t - wektor wyników komórki w danym takcie
- f_t - wektor sterujący bramką zapominającą
- g_t - wektor wartości generowanych przez warstwę główną
- i_t - wektor sterujący bramką wejściową
- o_t - wektor sterujący bramką wyjściową



Rysunek 1.6: Schemat komórki LSTM

Obliczenia wykonywane przez komórkę LSTM:

- $i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$
- $f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$
- $o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$
- $g_t = \tanh(W_{xg}x_t + W_{hg}h_{t-1} + b_g)$
- $c_t = f_t \otimes c_{t-1} + i_t \otimes g_t$

- $y_t = h_t = o_t \otimes \tanh(c_t)$

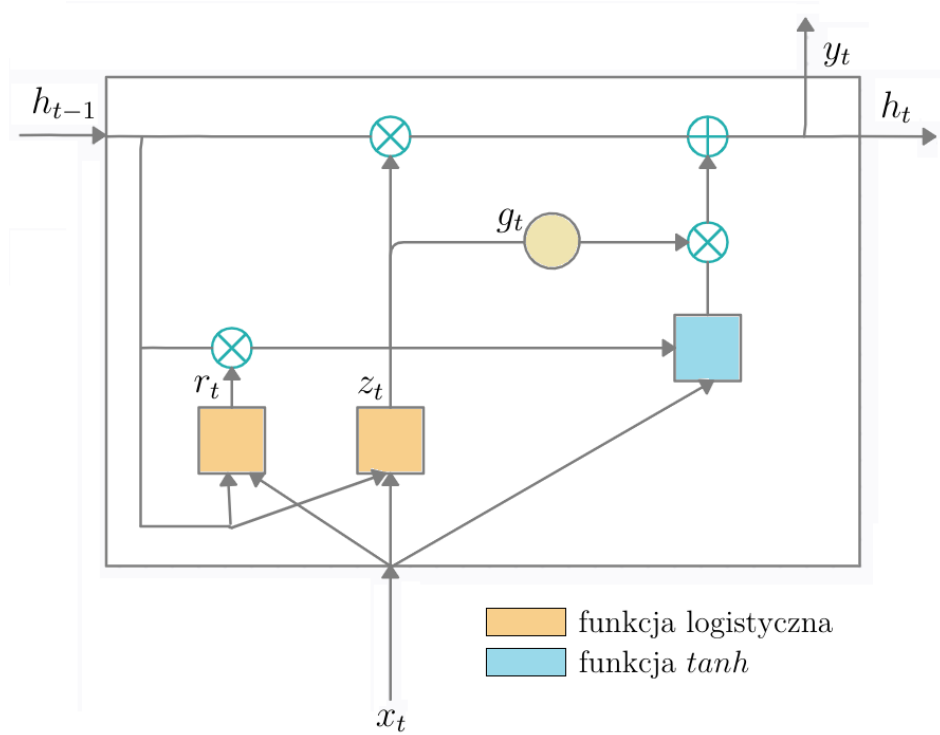
W równaniu przyjęto następujące oznaczenia:

- $W_{xi}, W_{xf}, W_{xo}, W_{xg}$ - macierze wag dla określonej warstwy (odpowiednio: bramki wejściowej, zapominającej, wyjściowej oraz warstwy głównej),
- $W_{hi}, W_{hf}, W_{ho}, W_{hg}$ - macierze wag dla każdej warstwy w połączeniu ze stanem krótkotrwałym h_{t-1} ,
- b_i, b_j, b_o, b_g - obciążenia odpowiadających warstw.

1.2.3 Architektura sieci GRU

Komórka GRU ma prostszą architekturę w porównaniu do komórki LSTM. Nie korzysta ona z wektora stanów c_t - jest on zcalony z wektorem h_t . Komórka GRU zawiera jedynie dwie bramki - resetującą i aktualizującą. Bramka aktualizująca przyjmuje podobne funkcje jak zapominająca i wejściowa w sieci LSTM - kontroluje wartość pojawiającą się na wyjściu i w zależności od niej podejmuje odpowiednią akcję: jeśli jest to 1 - informacja jest przekazywana dalej, jeśli 0 - zapominana[9]. Obliczenia wykonywane przez komórkę GRU:

- $z_t = \sigma(W_{xz}x_t + W_{hz}h_{t-1} + b_z)$
- $r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r)$
- $g_t = \tanh(W_{xg}x_t + W_{hg}(r_t \otimes h_{t-1}) + b_g)$
- $h_t = z_t \otimes h_{t-1} + (1 - z_t) \otimes g_t$



Rysunek 1.7: Schemat komórki GRU

Oznaczenia przyjęte na schemacie:

- g_t - warstwa główna
- r_t - bramka resetująca
- z_t - bramka aktualizująca

Rozdział 2

Budowa modeli

W celu przeprowadzenia badań przygotowanych zostało kilka modeli dla szeregu czasowego zużycia energii elektrycznej dla Polski, Niemiec oraz Hiszpanii. Modele zostały zbudowane w podobny sposób, zatem omówione zostaną głównie kroki podjęte w związku z przygotowaniem ich dla jednego szeregu czasowego - danych z Polski.

2.1 Przygotowanie danych

Pierwszym krokiem przygotowania danych do analizy było wybranie ze zbioru danych kolejnych pomiarów dla Polski spoza innych państw europejskich. W badanym szeregu czasowym na 87648 pomiarów obciążeń sieci energetycznej dla Polski znaleziono 10 braków w danych, występujących w marcu każdego roku o godzinie 2.

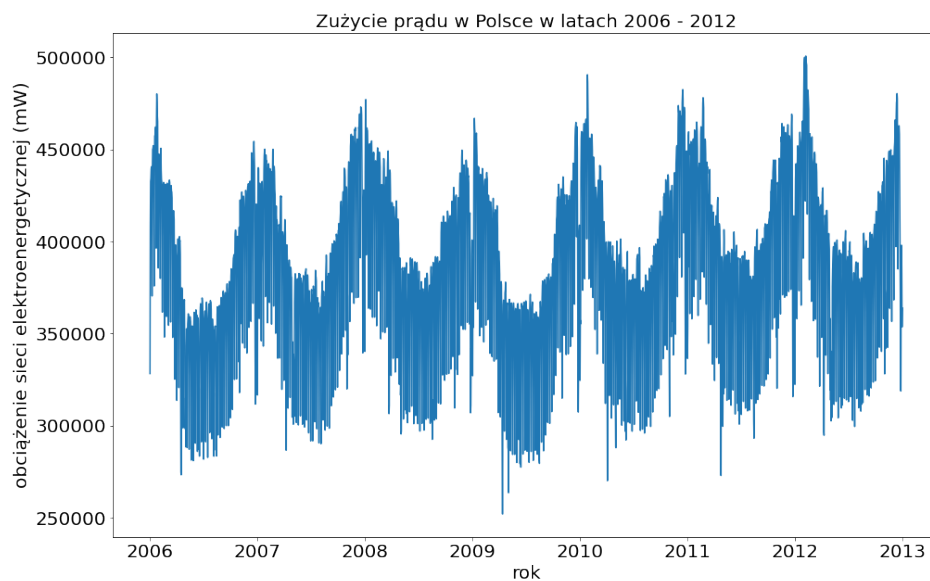
Out[55]:

	date	load	2
89496	2006-03-26	325238.0	NaN
89860	2007-03-25	319697.0	NaN
90231	2008-03-30	327246.0	NaN
90595	2009-03-29	307089.0	NaN
90959	2010-03-28	312067.0	NaN
91323	2011-03-27	333466.0	NaN
91687	2012-03-25	321402.0	NaN
92058	2013-03-31	298324.0	NaN
92422	2014-03-30	322001.0	NaN
92786	2015-03-29	342174.0	NaN

Rysunek 2.1: Znalezione braki danych

Są to brakujące godziny podczas zmiany czasu z zimowego na letni. Zostały one uzupełnione wartościami odpowiadającymi analogicznymi godzinom i dniom tygodnia w tygodniu poprzedzającym pomiar. Uzupełnienie braków w ten sposób zostało zmotywowane występowaniem silnej sezonowości w dziennym, tygodniowym i rocznym horyzoncie czasowym.

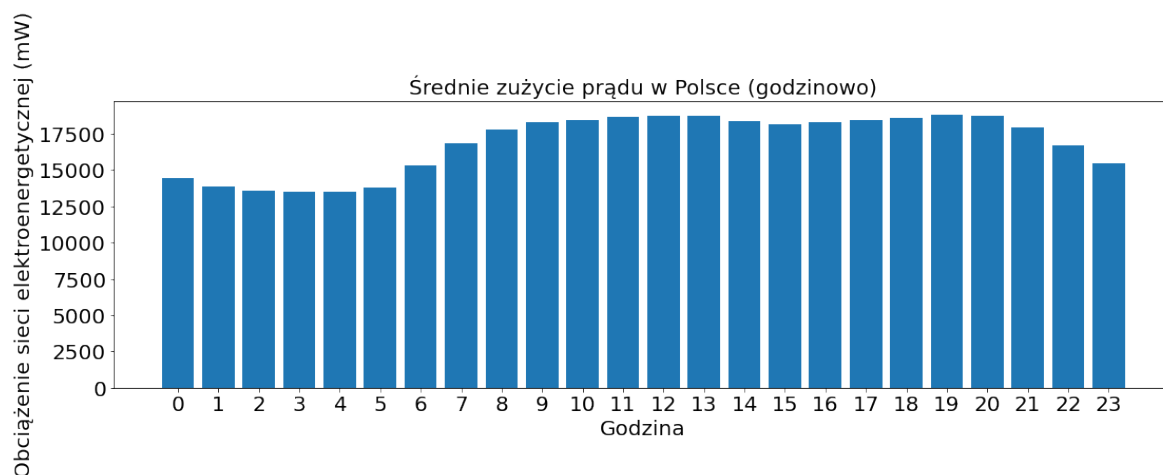
Dla ułatwienia dalszej analizy pobrana ramka danych została lekko zmodyfikowana - usunięte zostały niepotrzebne cechy, takie jak kraj. Trzy osobne kolumny zawierające informacje o roku, miesiącu oraz dniu wykonania pomiaru zostały złączone i przekonwertowane do formatu daty, która stała się identyfikatorem wiersza. W celu przeprowadzenia walidacji modelu, dane zostały podzielone na zbiór treningowy i testowy. W związku z potrzebą porównania predykcji dla różnych horyzontów czasowych zaczynających się w tym samym czasie, zbiór testowy i treningowy różni się w zależności od testowanych modeli. W przypadku statystycznego modelu ARMA, zbiór testowy i treningowy obejmuje odpowiednio daty 01.01.2006 - 26.01.2013 oraz 27.01.2013 - 31.12.2015. W przypadku sieci neuronowych, data podziału zbiorów jest przesunięta wstecz o długość sekwencji uczących $n = [7, 14, 21, 28]$. Predykcje zawsze rozpoczynają się dnia 27.01.2013r.



Rysunek 2.2: Szereg czasowy obciążeń sieci elektroenergetycznej w Polsce w latach 2006 - 2012

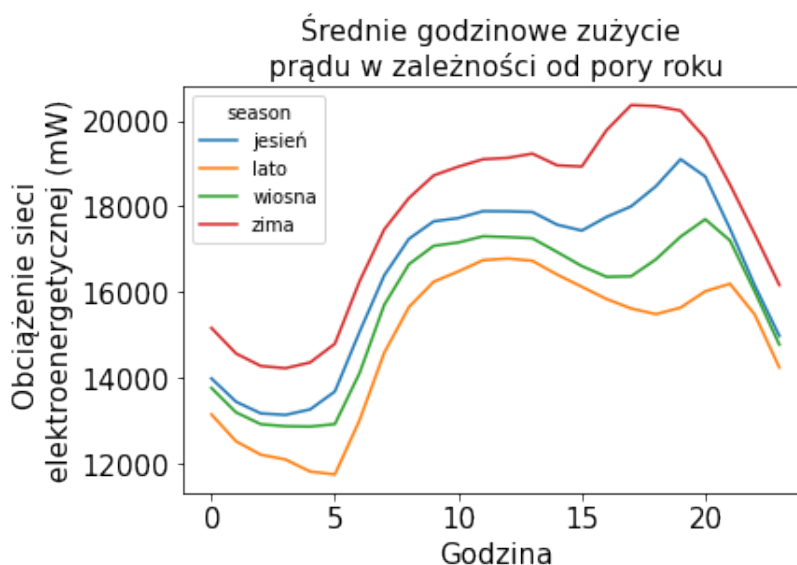
2.2 Eksploracja danych

W celu zobrazowania zależności obciążenia sieci elektroenergetycznej od czasu, dane zostały pogrupowane na kilka sposobów. Analizując wykres słupkowy średniego zużycia prądu w Polsce w zależności od godziny można łatwo zauważyć, że cykl dobowy wpływa znacząco na zapotrzebowanie energetyczne. Różnica wartości minimalnej i maksymalnej - odpowiednio dla godzin 3 (13507.1 mW) oraz 19 (18829.25 mW) wynosi aż 5322.149 mW.



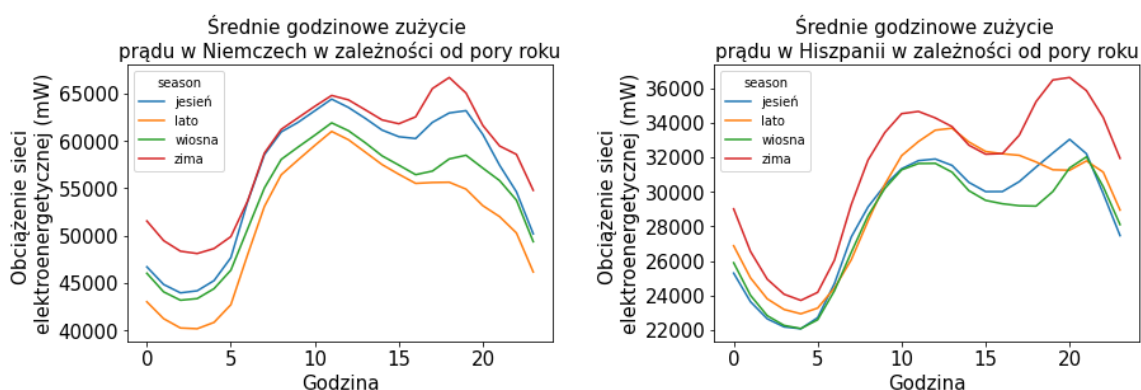
Rysunek 2.3: Wykres średniego zapotrzebowania na energię elektryczną w zależności od godziny w Polsce

Ze względu na potrzebę dodatkowego ogrzewania i częstszego oświetlania pomieszczeń w miesiącach zimowych i jesiennych, zdecydowano również przeanalizować wykresy średniego zużycia prądu w zależności od pory roku dla poszczególnych dni tygodnia oraz godzin w ciągu doby. W tym celu dane zostały podzielone na miesiące zimowe (grudzień, styczeń, luty), wiosenne (marzec, kwiecień, maj), letnie (czerwiec, lipiec, sierpień) oraz jesienne (wrzesień, październik, listopad). Na obu wykresach możemy zaobserwować silną zależność zużycia prądu od okresów ciepłych lub zimnych - przy nieznacznej zmianie kształtu wykresów, zapotrzebowanie na energię elektryczną wyraźnie wzrasta wraz ze zmianą czasu w roku na taki o niższej średniej temperaturze.



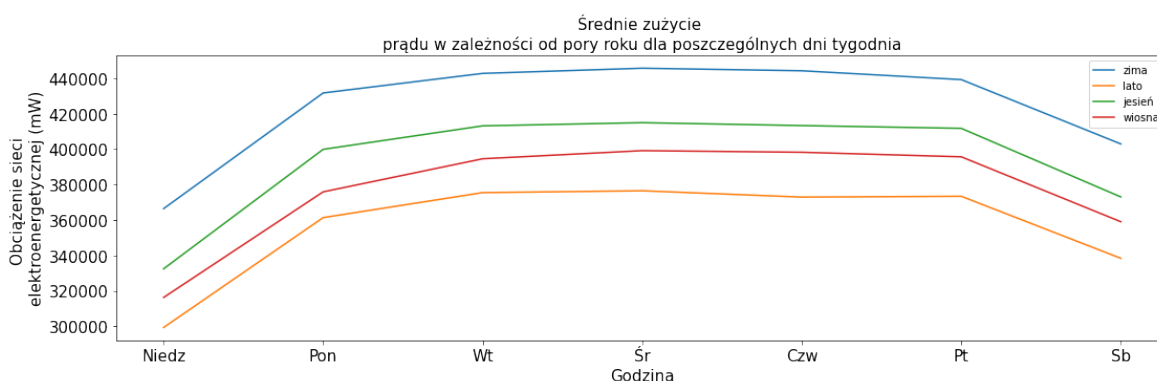
Rysunek 2.4: Wykres średniego zapotrzebowania na energię elektryczną w zależności od godziny w poszczególnych porach roku w Polsce

Dla porównania wygenerowano również wykresy obciążeń sieci elektroenergetycznej dla Hiszpanii i Niemiec. Jak można zauważyć, na cykl dobowy zapotrzebowania na energię elektryczną wpływa więcej czynników - ze względu na podobny tryb życia i temperatury, wykresy dla Polski oraz Niemiec mają bardzo podobny układ. Zapotrzebowanie dobowe w zależności od pory roku w Hiszpanii znacznie się jednak różni. Spowodowane jest to cieplejszym klimatem, mniejszą potrzebą ogrzewania domów w zimie lub jej całkowitym brakiem oraz siestą w ciągu dnia. Dużo większe zapotrzebowanie w okresie letnim jest typowe dla państw turystycznych. Dodatkowym obciążeniem sieci elektrycznej w lecie może być też działająca klimatyzacja podczas bardzo wysokich temperatur.



Rysunek 2.5: Wykres średniego zapotrzebowania na energię elektryczną w zależności od godziny w poszczególnych porach roku w Hiszpanii i w Niemczech

Nietrudno zauważyć również wyraźny spadek średniego obciążenia sieci elektroenergetycznej w weekend, przy czym najniższe wartości odnotowywane są standardowo w niedzielę, niezależnie od pory roku. Wiąże się to z obniżoną działalnością firm, spółek i zakładów przemysłowych pod koniec tygodnia.



Rysunek 2.6: Wykres średniego zapotrzebowania na energię elektryczną w zależności od dnia tygodnia w poszczególnych porach roku w Polsce

2.2.1 Wykresy pudełkowe

W celu przyjrzenia się wnikliwie innym zależnościom - takim jak rozproszenie oraz kształt rozkładu danych - pomiędzy obciążeniem sieci elektroenergetycznej w poszczególnych okresach w roku, wygenerowane zostały wykresy pudełkowe (ang. boxplot). Można odczytać z nich wiele istotnych informacji oraz porównać jak zmieniają się rozkłady danych podczas zmiany sposobu ich pogrupowania[10]. Wykresy pudełkowe tworzone są w oparciu o 5 wartości: kwartyli rzędu 1, 2 oraz 3 i minimalnej oraz maksymalnej typowej wartości rozkładu danych. Wąsami nazywamy linie łączące końce pudełka ze skrajnymi wartościami mieszczącymi się w zakresie typowych wartości[11]. Oblicza się je ze wzoru:

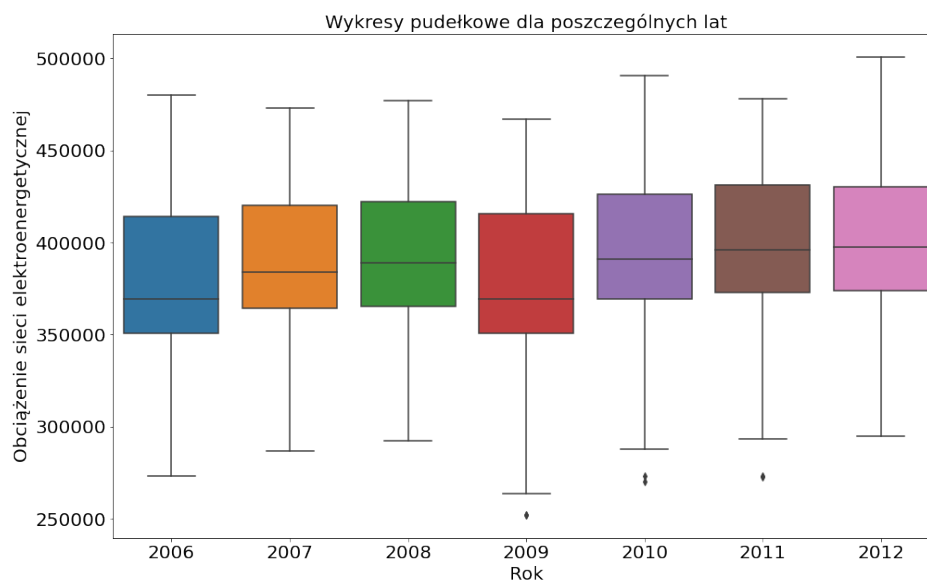
$$box_{min} = Q_1 - 1.5 * IQR \quad (2.1)$$

$$box_{max} = Q_3 + 1.5 * IQR \quad (2.2)$$

Długość pudełka reprezentuje wartość przedziału międzykwartylowego, którą matematycznie można zapisać jako:

$$IQR = Q_3 - Q_1 \quad (2.3)$$

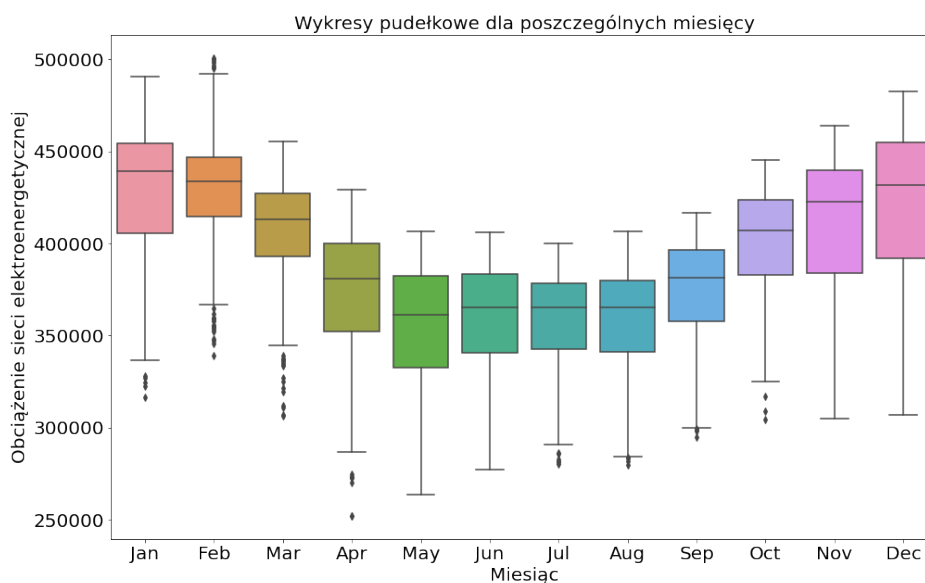
Linia umieszczona wewnątrz pudełka symbolizuje medianę, krawędzie pudełka - kwartył pierwszy (poniżej którego znajduje się 25% obserwacji) i trzeci (poniżej którego znajduje się 75% obserwacji). Poniżej krawędzi dolnego wąsa i powyżej górnego zaznaczane są wartości odstające.



Rysunek 2.7: Wykresy pudełkowe zapotrzebowania na energię elektryczną w zależności od roku w Polsce

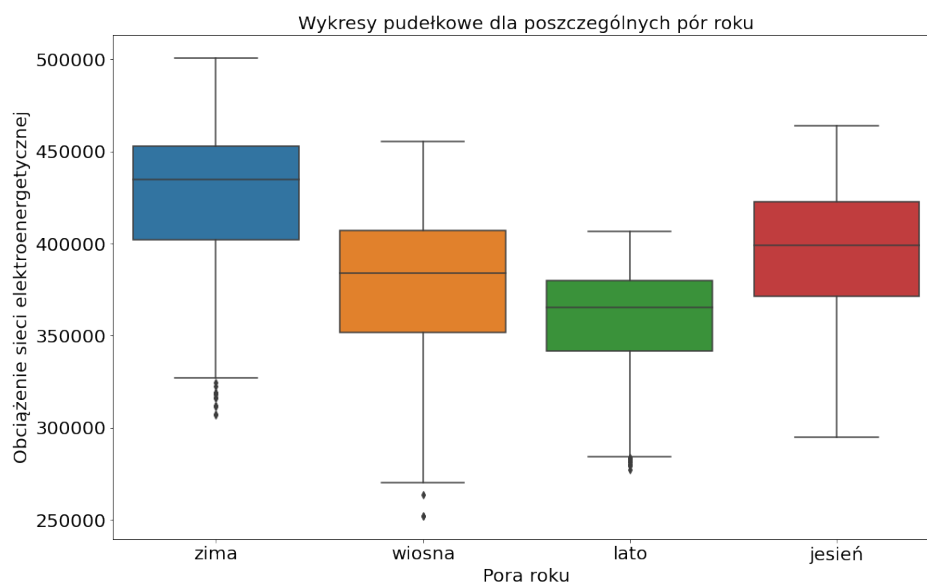
Pierwszym wygenerowanym wykresem jest wykres pudełkowy zapotrzebowania na energię elektryczną w zależności od roku. Widać na nim wyraźnie wzrost mediany wraz

z upływem czasu - wyjątkiem jest jedynie rok 2009. Może być to skutek generalnego rozwoju przemysłu w Polsce. Nie ma również wielu obserwacji odstających - pojawiają się jedynie w latach 2009 - 2011. Rozkłady badanej cechy nie są jednak symetryczne - dla całego zbioru treningowego pierwszy kwartył położony jest znacząco bliżej mediany, niż kwartył rzędu 3. Oznacza to, że rozkłady są asymetryczne prawostronnie.



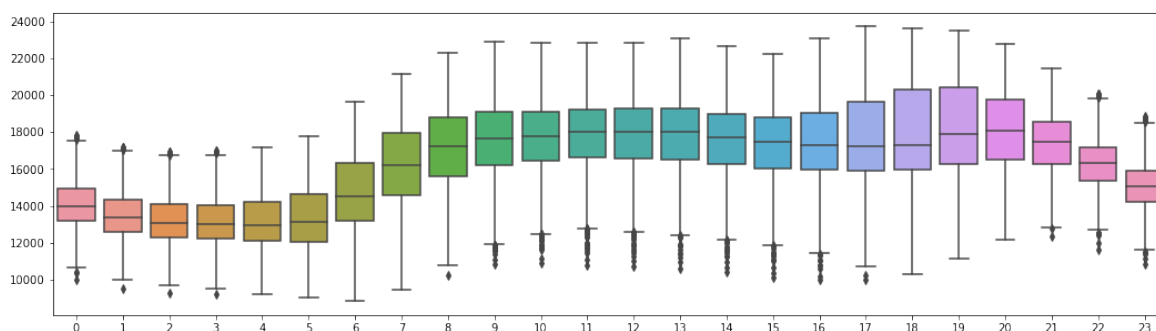
Rysunek 2.8: Wykresy pudełkowe zapotrzebowania na energię elektryczną w zależności od miesiąca w Polsce

Wykresy pudełkowe obciążenia sieci elektroenergetycznej w zależności od badanego miesiąca są dużo bardziej zróżnicowane. Pojawia się też wiele obserwacji odstających - szczególnie w okresie od stycznia do kwietnia - co może być czynnikiem utrudniającym dokładne predykcje w tych okresach. Rozkłady te są asymetryczne lewostronnie, w przeciwieństwie do rozkładu cechy podczas analizy rocznej. Wynika to z silnego wzrostu wartości mediany oraz kwartyli Q_1 i Q_3 podczas miesięcy zimniejszych. Na podobnym poziomie kształtują się jednak rozkłady miesięcy letnich - w przedziale czasowym od czerwca do sierpnia wartości Q_1 , Q_2 oraz Q_3 są do siebie bardzo zbliżone, a dane w tych miesiącach mniej rozproszone. Duże rozproszenie danych występuje natomiast podczas miesięcy zimowych i wiosennych.



Rysunek 2.9: Wykresy pudełkowe zapotrzebowania na energię elektryczną w zależności od pory roku w Polsce

Wykresy pudełkowe wygenerowane dla poszczególnych pór roku dobrze ukazują zależność zauważoną podczas analizy rozkładów dla miesięcy - niewielkie rozproszenie danych z okresu letniego, osiągające najniższe wartości. Rozkłady te również są asymetryczne lewostronnie, co jest zgodne z przewidywaniami.



Rysunek 2.10: Wykresy pudełkowe zapotrzebowania na energię elektryczną w zależności od godziny w Polsce

Analizując wykresy pudełkowe wygenerowane dla poszczególnych godzin w ciągu doby, dostrzega się wyraźną tendencję do spadku zapotrzebowania w godzinach nocnych - rozrzut wartości jest o wiele mniejszy, pojawiają się jednak obserwacje odstające. W przeciwieństwie do rozkładów dla miesięcy nie cechują się dużą asymetrycznością.

2.2.2 Współczynnik zmienności

Współczynnik zmienności to stosunek odchylenia standardowego i średniej[12]. Im wyższy współczynnik zmienności, tym wyższy jest stopień rozproszenia danych w populacji lub wśród danej cechy. Niski współczynnik zmienności można zaobserwować natomiast dla danych jednorodnych. Określa się go wzorem:

$$CV = \frac{\sigma}{\mu}, \quad \text{gdzie:} \quad (2.4)$$

- σ - odchylenie standardowe danej próby
- μ - średnia próby

0	1	2	3	4	5	6	7
9.191	9.495	9.682	9.982	11.027	12.552	14.507	15.064
8	9	10	11	12	13	14	15
14.161	12.944	12.037	11.97	12.001	12.214	12.381	12.861
16	17	18	19	20	21	22	23
14.606	15.461	15.203	14.15	11.918	9.407	8.375	8.696

Tabela 2.1: Współczynnik zmienności dla poszczególnych godzin

styczeń	luty	marzec	kwiecień	maj	czerwiec
8.890	8.065	8.174	10.552	9.652	9.128
lipiec	sierpień	wrzesień	październik	listopad	grudzień
8.2967	8.932	8.208	8.14	9.524	10.478

Tabela 2.2: Dzienny współczynnik zmienności dla poszczególnych miesięcy

wiosna	lato	jesień	zima
10.908	8.777	9.506	9.234

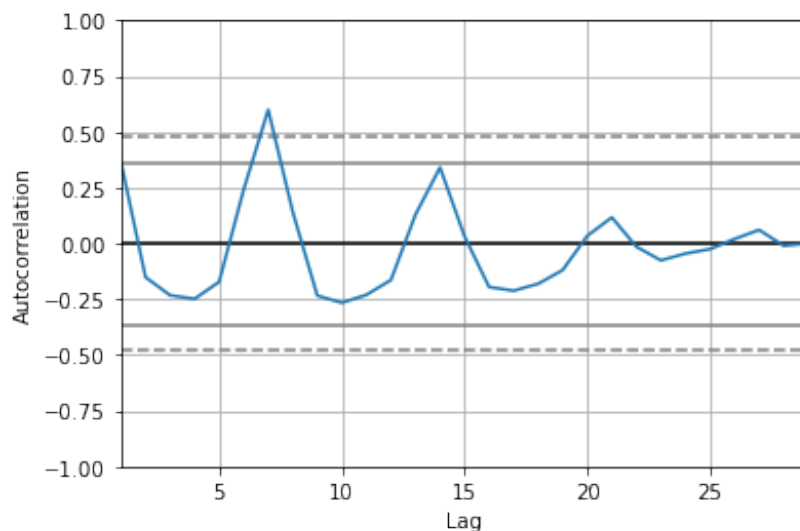
Tabela 2.3: Dzienny współczynnik zmienności dla poszczególnych pór roku

2006	2007	2008	2009	2010	2011	2012
11.961	11.099	10.756	12.138	11.624	10.8075	11.323

Tabela 2.4: Współczynnik zmienności dla poszczególnych lat

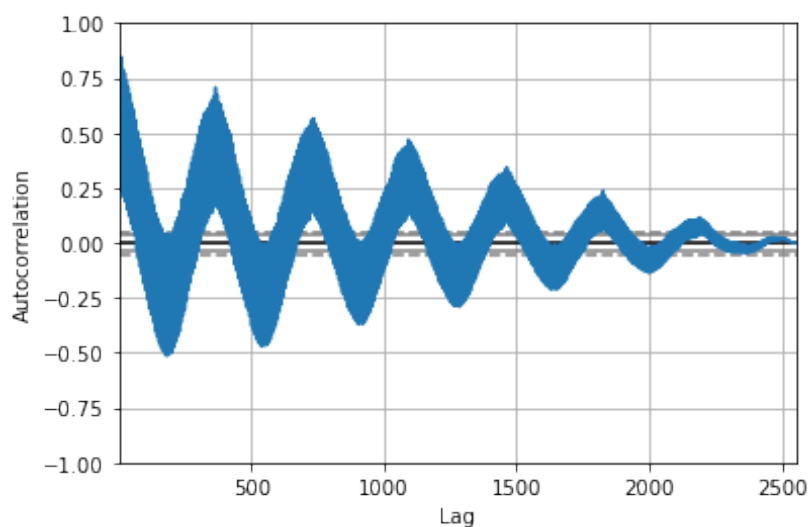
2.2.3 Autokorelacja

W celu analizy sezonowości skorzystano z autokorelacji szeregu czasowego. Zgodnie z założeniami, w krótkim przedziale czasowym bardzo silnie związane są ze sobą wartości obciążenia sieci energetycznej w odstępach 7-dniowych (i kolejno: 14, 21 oraz 28).

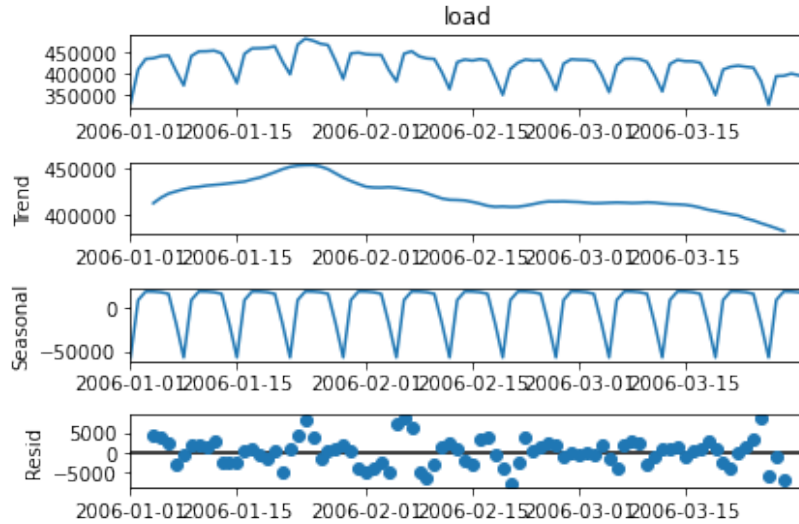


Rysunek 2.11: Autokorelacja dla krótkich wartości opóźnień (1 - 30 dni) dla Polski

Na wykresie autokorelacji obserwacji w dłuższym okresie czasu widać wyraźnie dodatnią korelację między wartościami z opóźnieniem równym około roku. Ujemna autokorelacja występuje natomiast dla okresów oddalonych od siebie o pół roku, co zgadza się z wcześniejszymi wnioskami.



Rysunek 2.12: Autokorelacja dla całego zbioru testowego dla Polski



Rysunek 2.13: Wykres dekompozycji szeregu czasowego obciążeń sieci elektroenergetycznej dla Polski (90 dni)

2.3 Metryki oceny jakości modeli

W celu oceny przydatności i jakości uzyskanych modeli, porównano następujące miary błędów:

- RMSE (ang. *root mean square error*) - pierwiastek błędu średniokwadratowego - mierzy, jak duża jest różnica pomiędzy wartościami rzeczywistymi a wartościami przewidzianymi przez model. Ze względu na podniesienie błędu do kwadratu przed jego uśrednieniem, RMSE nadaje dużą wagę obserwacjom odstającym - warto analizować jego wartość zwłaszcza wtedy, gdy duże błędy są szczególnie niepożądane. Określa się go wzorem:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2.5)$$

- MAE (ang. *mean absolute error*) - średni błąd bezwzględny - ze względu na brak potęgi we wzorze, jest dużo łatwiejszy do interpretacji w porównaniu do RMSE a jego wartość zawsze będzie równa lub mniejsza od niego. Im różnica między RMSE i MAE jest większa, tym większa jest wariancja błędów w próbie.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.6)$$

- MAPE (ang. *mean absolute percentage error*) - średni procentowy błąd absolutny - ze względu na procentową formę jest szczególnie przydatny podczas porównywania błędów predykcji szeregów czasowych o wyrażeniu różnych średnich wartościach:

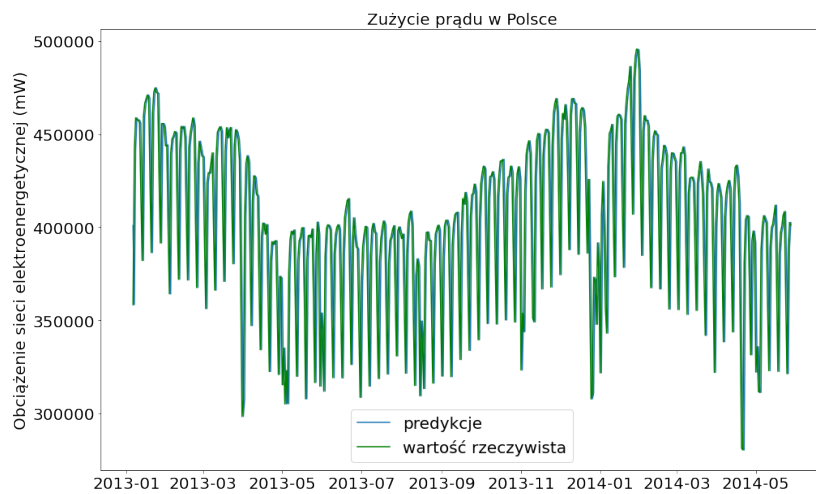
$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (2.7)$$

- ME (ang. *mean error*) - błąd średni - wskazuje, czy predykcje są zazwyczaj przeszacowane (gdy ME jest ujemne), czy niedoszacowane (gdy ME jest dodatnie). ME = 0 oznacza, że średnio wartości predykowane są takie same jak rzeczywiste realizacje szeregu czasowego.

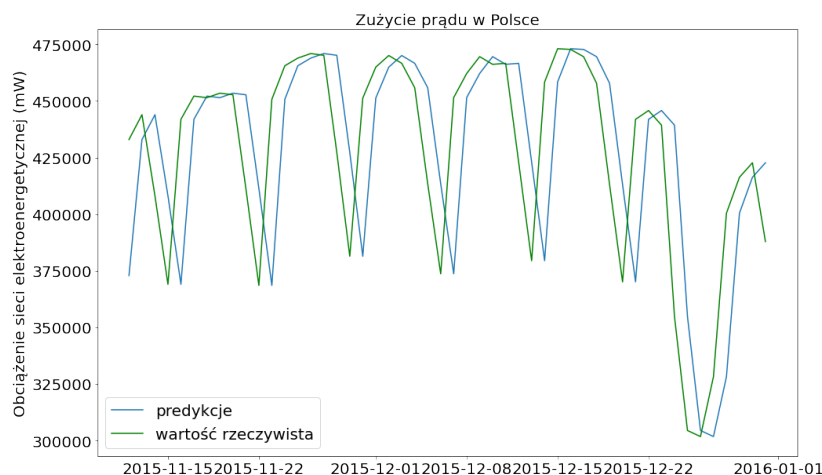
$$ME = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i) \quad (2.8)$$

2.3.1 Modele naiwne

W celu zbadania sensowności rozpatrywanych modeli, zostały utworzone również dwa proste modele naiwne. Ze względu na dość zbliżone wartości między dniami następującymi po sobie, pierwszy z nich przyporządkowuje wartości dla predykowanego dnia poprzednią wartością obciążenia elektrycznego.

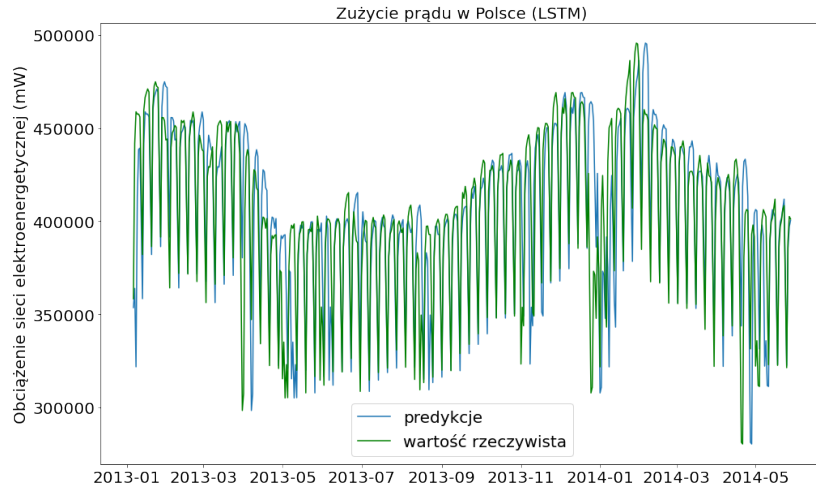


Rysunek 2.14: Wizualizacja modelu powtarzającego ostatnią predykcję dla całego zbioru testowego

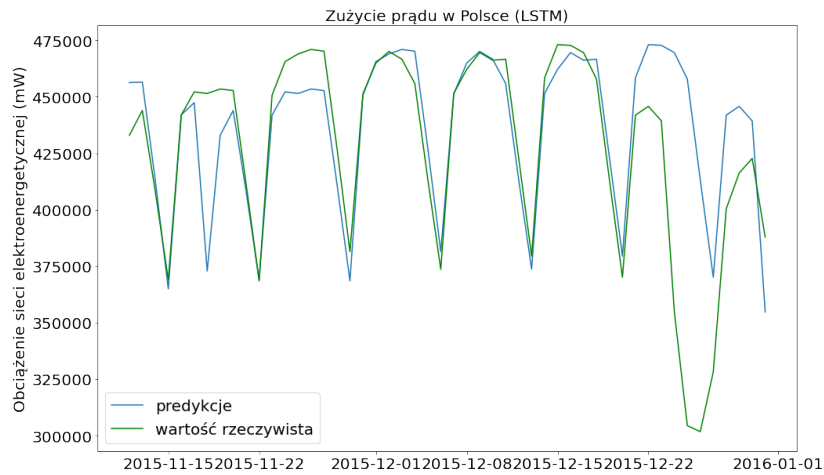


Rysunek 2.15: Wizualizacja modelu powtarzającego ostatnią predykcję dla ostatnich 50 dni ze zbioru testowego

Drugi natomiast - ze względu na sezonowość tygodniową - dla konkretnego punktu w czasie wybiera wartość odpowiadającą temu samemu dniowi tygodnia z tygodnia poprzedzającego.



Rysunek 2.16: Wizualizacja modelu powtarzającego predykcję sprzed tygodnia dla całego zbioru testowego



Rysunek 2.17: Wizualizacja modelu powtarzającego predykcję sprzed tygodnia dla ostatnich 50 dni ze zbioru testowego

2.4 Modele ARMA oraz ARIMA

Definicja 2.1 (Stacjonarność szeregu czasowego [2])

Szereg czasowy $\{X_t, t \in \mathbb{Z}\}$, gdzie $\mathbb{Z} = \{0, \pm 1, \pm 2, \dots\}$ nazywamy stacjonarnym, gdy są spełnione następujące warunki:

- $E|X_t|^2 < \infty$ dla każdego $t \in \mathbb{Z}$,
- $EX_t = m$ dla każdego $t \in \mathbb{Z}$, $m = \text{const}$

- $\gamma_X(r, s) = \gamma_X(r + t, s + t)$ dla każdego $r, s, t \in \mathbb{Z}$, gdzie $\gamma_X(r, s)$ - funkcja autokowariancji szeregu X_t

Definicja 2.2 (Biały szum [2])

Proces $\{Z_t\}$ nazywamy białym szumem ze średnią 0 i wariancją σ^2

$$\{Z_t\} \sim WN(0, \sigma^2), \quad (2.9)$$

wtedy i tylko wtedy, gdy ma średnią 0 oraz funkcję kowariancji określoną wzorem:

$$\gamma(h) = \begin{cases} \sigma^2, & \text{gdy } h = 0, \\ 0, & \text{gdy } h \neq 0 \end{cases} \quad (2.10)$$

Definicja 2.3 (Różnicowanie)

Różnicowanie jest metodą transformacji danych w celu uzyskania stacjonarnej postaci szeregu czasowego. Opiera się na przekształceniu kolejnych wartości obserwacji za pomocą równania:

$$d_t = x_t - x_{t-1} \quad (2.11)$$

W przypadku różnicowania sezonowego, operacja przebiega analogicznie:

$$d_t = x_t - x_{t-n}, \quad (2.12)$$

gdzie n - wartość opóźnienia czasowego przypadająca na sezon.

Definicja 2.4 (Proces ARMA(p, q) [2])

Proces $\{Z_t\}, \{X_t, t = 0, \pm 1, \pm 2, \dots\}$ nazywamy procesem ARMA(p, q) jeśli $\{X_t\}$ jest stacjonarny i dla każdego t , zachodzi:

$$X_t - \phi_1 X_{t-1} - \dots - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \dots + \theta_q Z_{t-q}, \quad (2.13)$$

gdzie $\{Z_t\} \sim WN(0, \sigma^2)$. Mówimy, że $\{X_t\}$ jest procesem ARMA(p, q) ze średnią μ jeśli $\{X_t - \mu\}$ jest procesem ARMA(p, q).

Z powodu założenia, że proces ARMA(p, q) jest stacjonarny, przed użyciem go do analizy szeregu czasowego konieczne jest sprowadzenie go do odpowiedniej postaci - należy usunąć z danych zarówno trend, jak i sezonowość. W końcowym etapie, aby otrzymać ostateczne predykcje, trend i sezonowość są ponownie dodawane do modelu.

ARIMA(p, d, q) (ang. *autoregressive integrated moving average*) jest połączeniem procesu autoregresji AR(p), różnicowania I(d) oraz średniej ruchomej MA(q). Model ARIMA stosowany jest do niestacjonarnych procesów, w których trend można usunąć za pomocą różnicowania. Wartość parametru d - stopnia różnicowania - to wartość stopnia wielomianu dopasowanego do trendu w danych[13]. Proces ARMA jest procesem ARIMA z rzędem różnicowania $d = 0$.

W celu sprawdzenia, czy wektor danych jest stacjonarny, wykorzystuje się następujące testy statystyczne:

- test ADF (ang. *Augmented Dickey–Fuller test*) [15] - test rozszerzony Dickeya–Fullera poprzez badanie, czy w szeregu czasowym istnieje pierwiastek jednostkowy, sprawdza jego stacjonarność. Określa się w nim hipotezy:
 - H_0 - pierwiastek jednostkowy istnieje w badanym szeregu czasowym
 - H_1 - brak pierwiastka jednostkowego w badanym szeregu, co wskazuje na stacjonarność względem trendu szeregu czasowego.

Czym niższa statystyka testu, tym mamy silniejsze przesłanki do odrzucenia hipotezy zerowej na wybranym poziomie ufności α .

- test (KPSS) (Kwiatkowski–Phillips–Schmidt–Shin test) [17] - W celu przeprowadzenia testu, szereg rozpatrywany jest jako suma błędzenia losowego, trendu deterministycznego oraz błędu stacjonarnego:

$$y_t = \xi t + r_t + \epsilon_t, \quad \text{gdzie :}$$

- y_t , $t = 1, 2, \dots, T$ - obserwacje szeregu czasowego,
- t - trend deterministyczny,
- r_t - proces błędzenia losowego,
- ϵ_t - błąd stacjonarny obserwacji.

Hipotezy testu KPSS określone są następująco:

- H_0 - badany szereg jest stacjonarny względem trendu
- H_1 - badany szereg nie jest stacjonarny

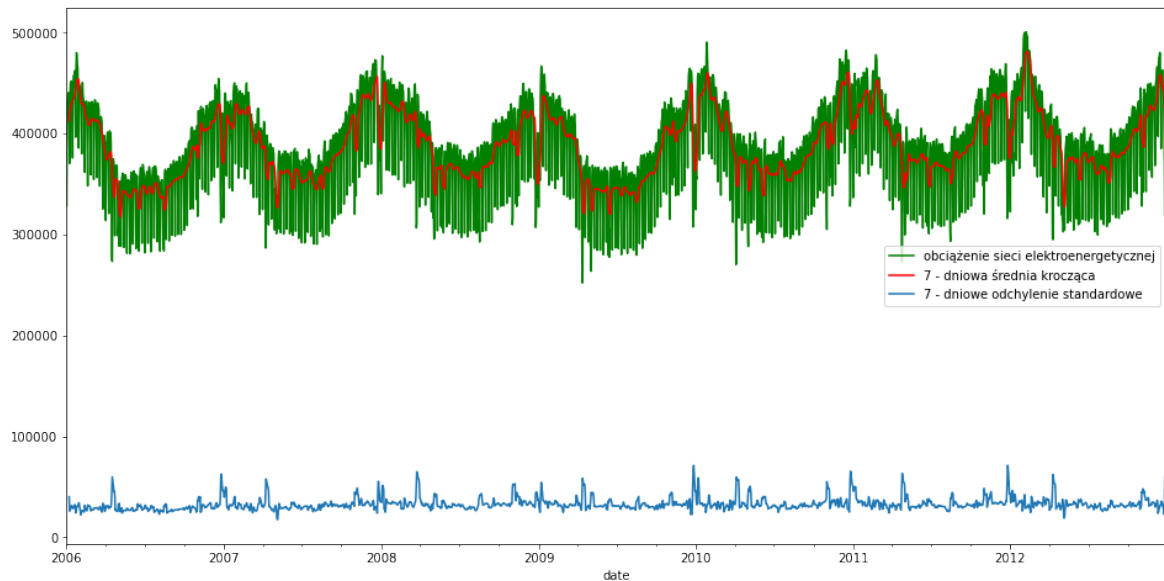
Wartości krytyczne dla statystyki testowej KPSS:

wartość krytyczna	0.10	0.05	0.025	0.01
statystyka KPSS (z trendem)	0.119	0.146	0.176	0.216
statystyka KPSS (bez trendu)	0.347	0.463	0.574	0.739

Tabela 2.5: Dzienny współczynnik zmienności dla poszczególnych miesięcy

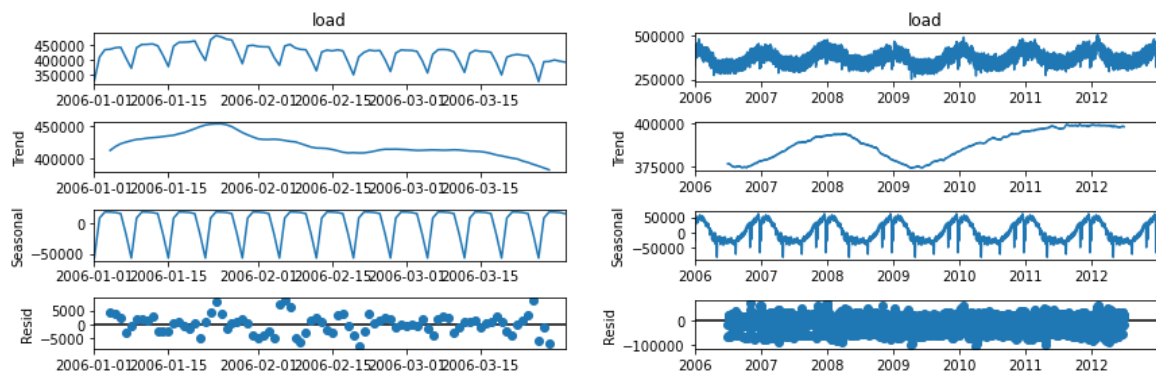
2.4.1 Dobór parametrów modelu

W celu lepszego zobrazowania trendu występującego w danych, zdecydowano się przeanalizować wykres przedstawiający średnią kroczącą z 7 dni predykcji wraz z odchyleniem standardowym, dzięki któremu można zobaczyć, jak bardzo wartości oddalone są od średniej.



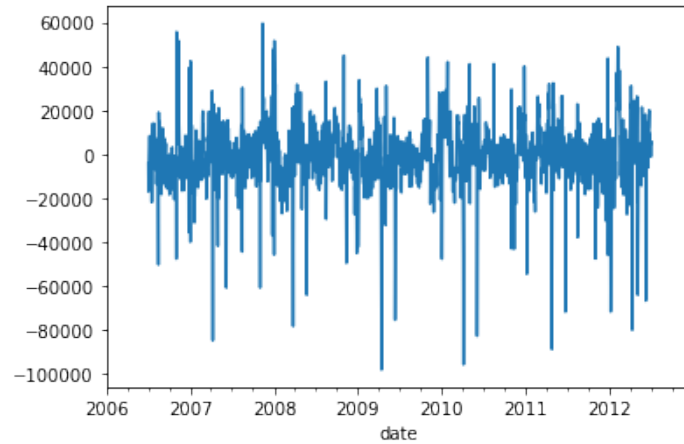
Rysunek 2.18: Wykres 7-dniowej średniej kroczącej oraz odchylenia standardowego dla szeregu czasowego obciążenia sieci elektroenergetycznej dla Polski

Jednym z podstawowych założeń, które należy spełnić przed zastosowaniem modelu ARMA jest osiągnięcie stacjonarności szeregu. Pierwszym krokiem w przygotowaniu danych do przetworzenia przez model jest usunięcie sezonowości. W szeregach czasowych obciążenia dziennego sieci elektroenergetycznej możemy zaobserwować występowanie dwóch sezonowości: tygodniowej oraz rocznej - zmienność dobową nie jest analizowana. Standardowym podejściem mającym na celu podzielenie szeregu czasowego na poszczególne komponenty jest zastosowanie dekompozycji[21]. Dekompozycje szeregu na trend, sezonowość oraz residua dla okresów 7-dniowych oraz rocznych prezentują się następująco:



Rysunek 2.19: Dekompozycje szeregu czasowego obciążeń sieci elektroenergetycznej dla Polski; kolejno dla okresów: 7 - dniowego oraz 365 - dniowego

Łatwo zauważyć, że wskazana na wykresie linia trendu dla okresu 7 - dniowego zachowuje nadal sezonowość roczną. Zakładając, że odjęcie od badanego szeregu czasowego składowej sezonowości dla okresu 365 dni usunie również wskazany trend, zdecydowano w następnym kroku odjąć jedynie trend dla okresu rocznego.



Rysunek 2.20: Szereg czasowy danych obciążeń sieci elektroenergetycznej dla Polski po zlikwidowaniu sezonowości tygodniowej oraz rocznej oraz odjęciu rocznego trendu

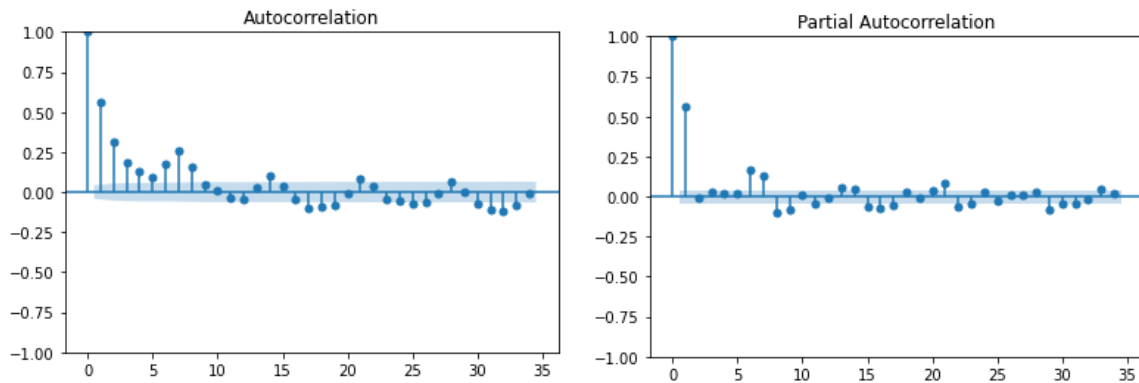
Przeprowadzono rozszerzony test Dickeya - Fullera oraz test KPSS, aby potwierdzić stacjonarność otrzymanego szeregu:

statystyka ADF: -9.463414	statystyka KPSS: 0.041691
p-wartość: 0.000000	p-wartość: 0.100000

Rysunek 2.21: Statystyki testowe testów stacjonarności szeregu

Ze względu na p -wartość $< 0,5$ dla testu ADF przy niskiej wartości statystyki testowej oraz p -wartość $> 0,5$ przy statystyce testowej KPSS mieszczącej się w przedziale $[0.347, 0.463]$, przyjmujemy hipotezę o stacjonarności rozkładu.

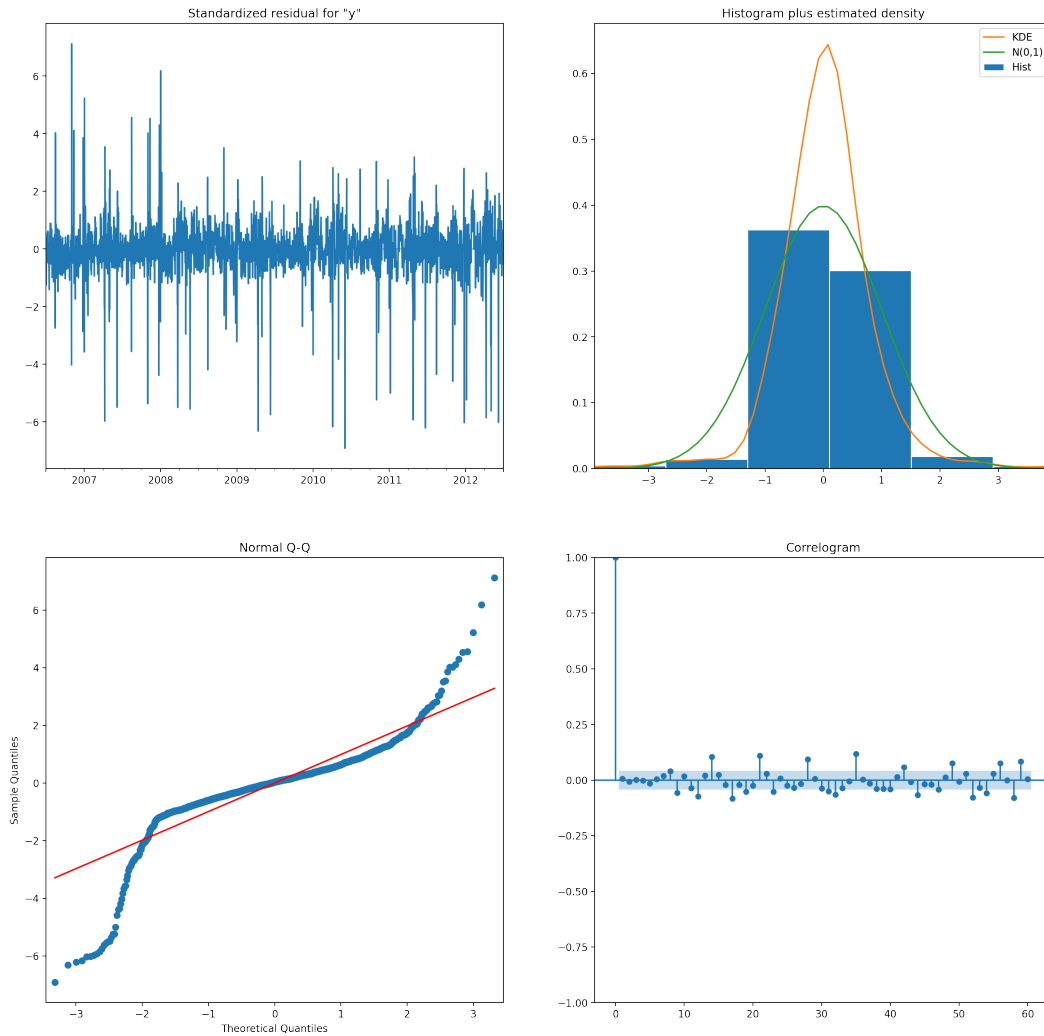
W celu doboru odpowiednich parametrów wstępnych dla modelu ARIMA posłużono się wykresem autokorelacji. Ze względu na stacjonarność szeregu parametr różnicowania $d = 0$.



Rysunek 2.22: Wykresy autokorelacji i autokorelacji częściowej szeregu czasowego po usunięciu trendu rocznego oraz sezonowości

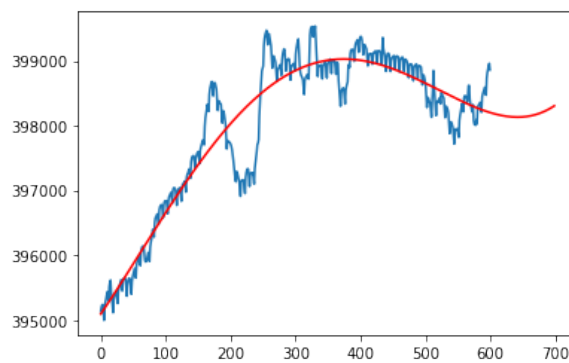
Wartości bezwzględne zarówno autokorelacji jak i autokorelacji częściowej maleją dla kolejnych opóźnień, a kształt wykresów wskazuje na występowanie obu pozostałych parametrów. Uznano, że parametr średniej ruchomej q modelu $ARMA(p, q)$ przyjmuje wartość $q = 7$, ze względu na ilość opóźnień z istotnymi wartościami na wykresie autokorelacji ACF. Parametr autoregresyjny dla modelu ustawiono na $p = 1$, ze względu na występowanie tylko jednego istotnego opóźnienia na wykresie autokorelacji częściowej.

Poniżej znajdują się wykresy diagnostyczne reszt uzyskanych z tak dobranego modelu. Histogram oraz wykres kwantyl-kwantyl pokazują, że residua mają rozkład leptokurtyczny. Charakteryzuje się on wyższym skupieniem obserwacji wokół wartości średniej niż ma to miejsce w przypadku rozkładu normalnego. Wartości błędów nie są jednak istotnie skorelowane ze sobą. Możemy uznać, że reszty są białym szumem i model jest poprawny .



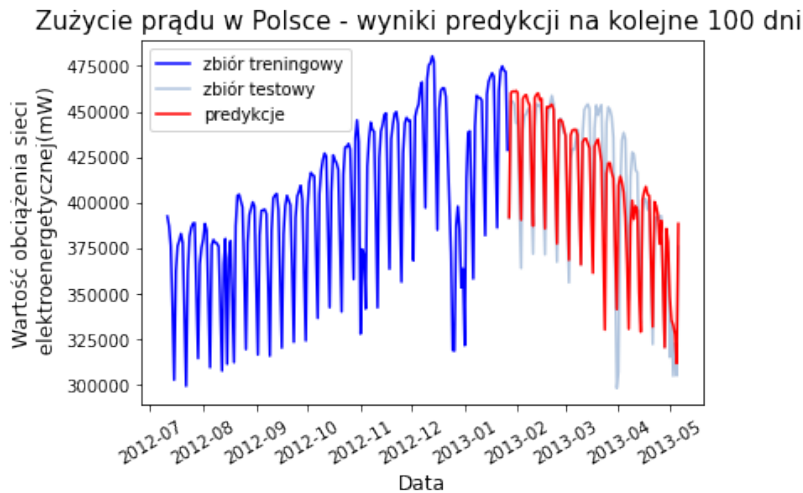
Rysunek 2.23: Wykresy diagnostyczne residuów rozpatrywanego modelu

W celu uzyskania ostatecznych predykcji, należy dodać odjęty trend i sezonowości badanego szeregu. Został on przybliżony wielomianem, aby wychwycić kolejne wartości składowej trendu:



Rysunek 2.24: Wykres wielomianu używanego do dalszej prognozy trendu

Predykcje uzyskane po dodaniu wszystkich składowych modelu prezentują się następująco:



Rysunek 2.25: Predykcje obciążeń sieci elektroenergetycznej na kolejne 100 dni dla Polski

Wyniki błędów RMSE dla różnych horyzontów czasowych zostały przedstawione w tabeli:

horyzont czasowy	7	14	30	60	90
RMSE	12872.09	13356.07	11384.24	15897.61	20664.14

Tabela 2.6: RMSE dla poszczególnych horyzontów czasowych dla modelu ARMA(7,1)

2.4.2 Kryteria informacyjne

W celu wybrania najlepszego modelu spośród modeli posiadających różną liczbę predyktorów, stosowane są kryteria informacyjne: Akaikego (AIC) oraz Schwarza (BIC). W oby przypadkach, im statystyka ma mniejszą wartość, tym model jest lepszy.

Kryterium informacyjne Akaikego AIC[19]

AIC (*Akaike Information Criterion*) zakłada wybór modelu minimalizującego:

$$-2 \log L_M + 2p_M, \quad \text{gdzie}$$

- L_M - funkcja wiarygodności dla modelu M ,
- p_M - liczba zmiennych w modelu
- $2p_M$ - kara za złożoność modelu \rightarrow bez tego członu otrzymalibyśmy model ze wszystkimi zmiennymi

Bayesowskie Kryterium informacyjne Schwarza BIC[20]

BIC (*Bayesian Information Criterion*) zakłada wybór modelu minimalizującego:

$$-2 \log L_M + p_M \log n, \quad \text{gdzie}$$

- L_M - funkcja wiarygodności dla modelu M ,
- p_M - liczba zmiennych w modelu
- $p_M \log n$ - kara za złożoność modelu \rightarrow bez tego członu otrzymalibyśmy model ze wszystkimi zmiennymi

Statystyka BIC nakłada zazwyczaj większą karę na modele zawierające wiele predyktorów, skutkuje więc wyborem prostszych modeli.

2.5 Modele sieci LSTM i GRU

2.5.1 Parametry modelu

Batch size jest hiperparametrem określającym liczbę próbek treningowych do przetworzenia przez model przed zaktualizowaniem jego wewnętrznych parametrów. Z uwagi na jego specyfikę, wielkość powinna być podzielna przez 8 (liczba rdzeni TPU). TPU (ang. Tensor Processing Unit) jest układem scalonym akceleratora AI opracowanym w celu uczenia sieci neuronowej przy użyciu wykorzystywanego przeze mnie pakietu TensorFlow. W modelu sieci LSTM batch size został ustawiony na 32.

Optymalizator ADAM

Optymalizator Adam został przedstawiony po raz pierwszy przez Diederika Kingmę oraz Jimmiego Ba w publikacji "Adam: A Method for Stochastic Optimization" [18]. Jest algorytmem, który można wykorzystać zamiast klasycznej stochastycznej procedury spadku wzdłuż gradientu do iteracyjnego aktualizowania wag sieci na podstawie danych uczących. Metoda wymaga tylko gradientów pierwszego rzędu, a szacowanie parametrów odbywa się przy pomocy obliczenia wartości dwóch pierwszych momentów gradientów.

2.5.2 Optymalizacja Bayesowska

Optymalizacja Bayesowska jest metodą wykorzystującą twierdzenie Bayesa do wyboru wartości hiperparametrów, które z dużym prawdopodobieństwem pozwolą osiągnąć lepszy wynik podczas trenowania modelu. Pomaga znacznie usprawnić proces nauki modelu w porównaniu do innych, prostszych sposobów, do których należą m.in. grid search oraz random search - algorytmy przeszukujące całą siatkę różnych konfiguracji parametrów oraz dobierające je w sposób losowy, kontrolując przy tym wybraną metrykę oceny jakości modelu. Algorytm wykorzystuje twierdzenie Bayesa w celu znalezienia maksimum lub minimum funkcji, której dziedziną jest przestrzeń hiperparametrów a zbiór wartości to wybrana metryka błędów[16, 13]. Posiłkuje się dodatkowo funkcją

akwizycji oraz funkcją zastępczą. Funkcja akwizycji to funkcja $a : X \rightarrow R$ używana do określenia, który punkt w zbiorze X jest kolejnym do obliczenia funkcji celu. Sprowadza się do problemu optymalizacji, gdzie $x_{n+1} = \operatorname{argmax}_x$, $x \in X$. Ocena funkcji $a(x)$ wykorzystuje jedynie model probabilistyczny, dzięki czemu jest dużo mniej kosztowna niż ocena samej funkcji celu. Funkcja zastępująca aproksymuje natomiast poprawność modelu dla danych hiperparametrów. Wykorzystywane jest twierdzenie Bayesa:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \quad \text{gdzie:} \quad (2.14)$$

- $P(A)$, $P(B)$ - prawdopodobieństwa zajścia odpowiednio zdarzeń A oraz B
- $P(A|B)$ - prawdopodobieństwo warunkowe zajścia zdarzenia B , gdy prawdziwe jest zdarzenie A

Kroki działania optymalizacji Bayesowskiej prezentują się następująco: [14]

1. Wykorzystanie funkcji akwizycji do wybrania odpowiedniej próbki
2. Wykorzystanie funkcji celu do oceny próbki
3. Aktualizacja funkcji zastępczej i zapisanie wyników
4. Powrót do punktu 1.

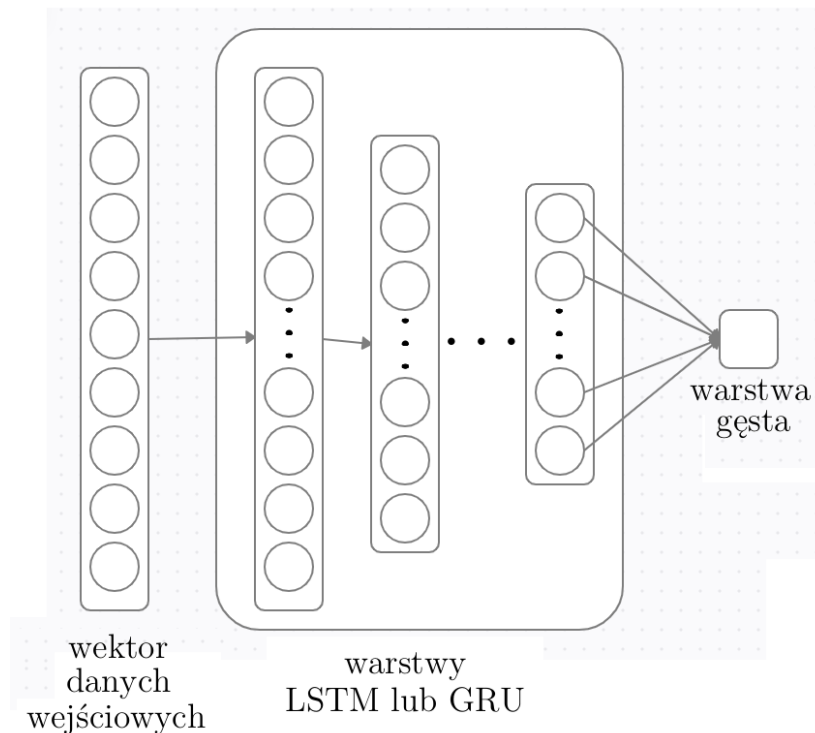
Maksymalna ilość prób dla optymalizacji Bayesowskiej w modelach sieci neuronowych została ustawiona na 30.

2.5.3 Wybór najlepszego modelu

W celu wyboru najlepszego modelu LSTM porównana została kombinacja różnych wartości parametrów. Dla każdego przypadku współczynnik uczenia został ustawiony na wartość 0.1, współczynnik batch size na 32, a maksymalna liczba epok na 1000. Wielkość sekwencji podawana w okienku do nauki predycji sieci została sprawdzona dla kilku pierwszych wielokrotności liczby 7 z uwagi na występowanie silnej sezonowości tygodniowej w oryginalnym szeregu: 7, 14, 21 oraz 28. Przed przetworzeniem szeregu przez sieć LSTM i GRU wektor danych wejściowych został ustandaryzowany: usunięto automatycznie trend oraz średnią stosując równanie $z = \frac{x-\mu}{\sigma}$, gdzie μ to średnia populacji, σ - odchylenie standardowe z populacji a x - wektor danych wejściowych. W wyniku takiego przekształcenia, wartość oczekiwana zmiennej losowej jest równa 0 a odchylenie standardowe 1.

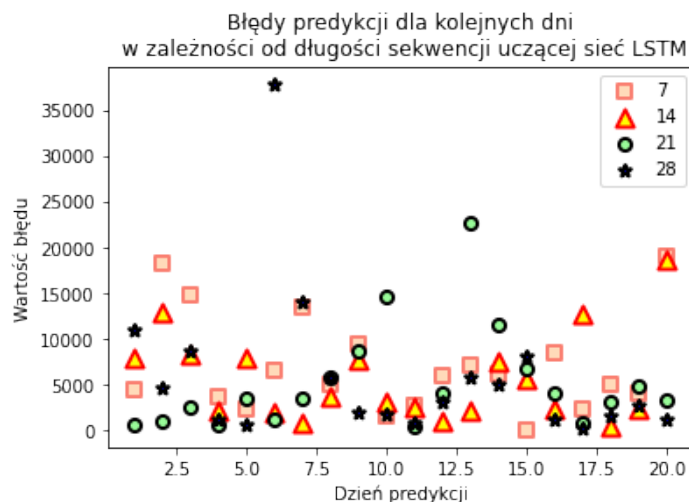
Sprawdzone zostały również różne modele sieci neuronowych - liczby neuronów oraz warstw sieci zostały wybrane tak, aby dawały jak najlepsze rezultaty, opierając się na minimalizacji błędów sieci. Badany model sieci jest modelem sekwencyjnym: złożony jest ze stosu połączonych ze sobą kolejno warstw. Optymalna liczba warstw wybrana została z wektora $n_{lay} = [1, 2, 3, 4, 5]$. Tworzona sieć neuronowa dla kolejnych warstw komórek LSTM lub GRU aż do warstwy przedostatniej przyporządkowywała ilość neuronów równą $n_{neurons} * (n_{lay} - 1)$. Ostatnia warstwa LSTM zawierała pełną liczbę neuronów. Ilość

neuronów $n_{neurons}$, która pozwala osiągać modelowi najlepsze wyniki wybrano przeszkukując wektor liczb od 10 do 120 ze skokiem co 4: $n_{neurons} = [10, 14, 18, \dots, 110, 114, 118]$. Ostatnią warstwą całej sieci jest warstwa gęsta (zawierająca połączenia neuronów typu każdy z każdym). Schemat budowy sieci prezentuje się następująco:



Rysunek 2.26: Ogólny schemat wybranego połączenia warstw sieci LSTM i GRU

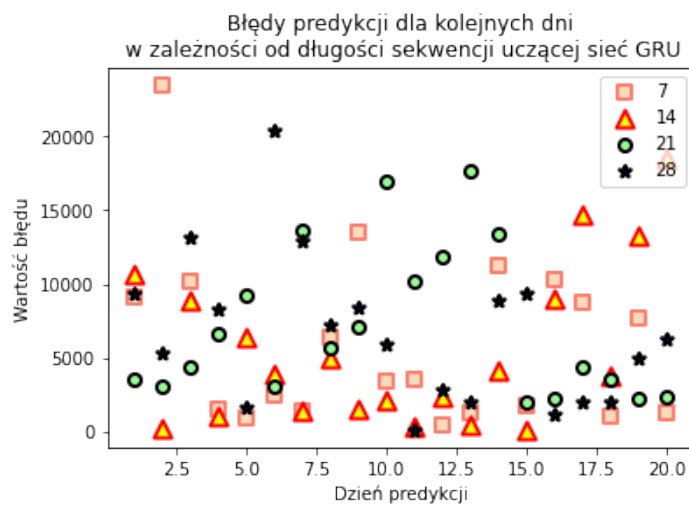
Za najlepszy model uznano ten, dla którego wartość metryki błędu $RMSE$ była najmniejsza.



Rysunek 2.27: Zależność błędów bezwzględnych predykcji od długości sekwencji uczącej

LSTM	sekwencja			
horyzont	7	14	21	28
7	22473,16	12715,16	13332,56	16304,89
14	16334,71	9880,50	9817,75	11869,90
30	11658,65	8391,44	7302,38	8820,70
60	9257,11	7979,87	6211,81	7193,65
90	9428,67	9569,08	7522,78	9332,54

Rysunek 2.28: RMSE dla poszczególnych długości sekwencji sieci LSTM w różnych horyzontach czasowych



Rysunek 2.29: Zależność błędów bezwzględnych predykcji od długości sekwencji uczącej

GRU	sekwencja			
horyzont	7	14	21	28
7	15107,66	12056,67	12823,83	11592,92
14	11357,05	8976,68	10108,49	9218,25
30	8280,31	6905,64	7599,54	7842,10
60	7157,45	5799,26	6124,59	6954,01
90	11625,78	8342,10	7447,03	8176,41

Rysunek 2.30: RMSE dla poszczególnych długości sekwencji sieci GRU w różnych horyzontach czasowych

Najniższe wartości błędów *RMSE* zostały w tabelach zaznaczone kolorem czerwonym. Za najefektywniejsze rozmiary sekwencji uczącej zostały uznane sekwencje o długościach:

- 21 - dla sieci opartej na komórkach LSTM
- 14 dla sieci opartej na komórkach GRU.

Rozdział 3

Rezultaty

Analiza szeregów czasowych 3 różnych państw wyraźnie pokazuje, że składa się na nie kilka złożonych komponentów powiązanych ze zmiennością szeregu w różnych okresach czasu. Wyróżniamy wśród nich:

- trend, tendencję długookresową
- wahania sezonowe
- ruchy cykliczne długoterminowe
- wahania nieregularne, niemożliwe do modelowania w sposób matematyczny

Wahania sezonowe i ruchy cykliczne długoterminowe są ściśle związane z oddziaływaniem klimatu i wpływem gospodarki, która również wykazuje tendencje do okresowych spadków oraz wzrostów[21]. Szczególnie widoczne jest to w przypadku państw, w których dużą rolę gospodarczą pełni turystyka.

Proste modele statystyczne takie jak ARMA mogą być stosowane do predykcji szeregów czasowych charakteryzujących się wielokrotną sezonowością po wykonaniu odpowiednich przekształceń. Ze względu na swoją liniową specyfikę oraz ograniczoną zdolność do dopasowania i modelowania wzorców sezonowych, mogą występować problemy z wyodrębnieniem długoterminowych tendencji[25]. Zakładają, że kształt wzorców sezonowych nie zmienia się w czasie, podczas gdy mogą występować subtelne różnice pomiędzy kształtem typowego tygodnia wiosennego, letniego, zimowego lub jesiennego. W związku z tym, prognozy długoterminowe mogą być dużo bardziej niedokładne niż te otrzymane w wyniku analizy za pomocą rekurencyjnych sieci neuronowych, komórek GRU i LSTM, które charakteryzują się większą uniwersalnością.

3.1 Wizualizacje predykcji poszczególnych modeli

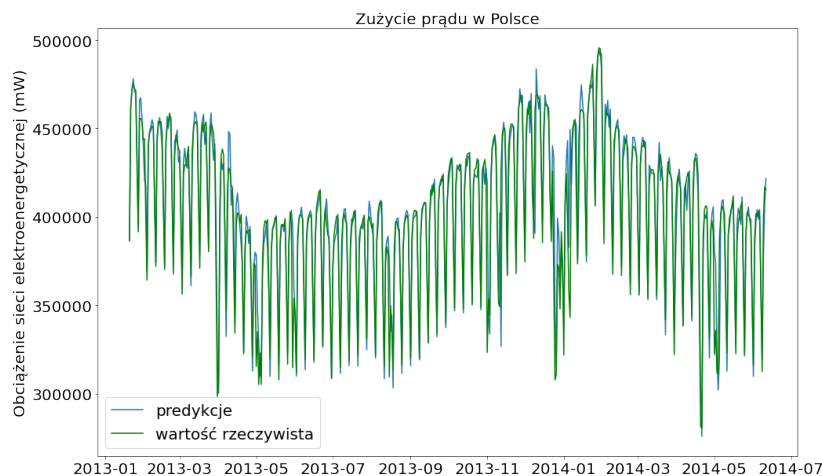
W celu wizualizacji dokładności przygotowanych modeli LSTM i GRU, wyniki zostały przedstawione na wykresach. Pierwszy z nich przedstawia porównanie predykcji oraz rzeczywistych wartości szeregu czasowego na całym zbiorze testowym. Z uwagi na duży rząd wielkości, różnice w takiej skali są mało widoczne, dlatego przedstawiony został również wykres dla 50 dni. Mimo, że jest to koniec zbioru testowego, możemy

zauważyć, że modele nadal radzą sobie bardzo dobrze i nie odbiegają mocno od wartości rzeczywistych.

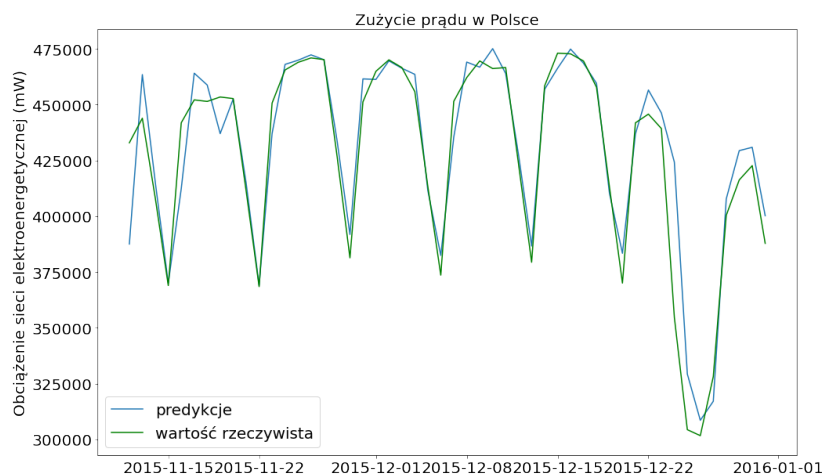
Wykresy 3.3 oraz 3.6 to wizualizacje historii treningu modeli, mówiące o zachowaniu *training loss* oraz *validation loss*. *Training loss* jest metryką używaną do oceny dopasowania modelu uczenia głębokiego do danych szkoleniowych - ocenia błąd modelu na zbiorze uczącym. *Training loss* oblicza się jako sumę błędów dla każdego przykładu w zbiorze uczącym. *Validation loss* jest miarą używaną do oceny wydajności modelu uczenia głębokiego w zbiorze walidacyjnym. Obliczana jest analogicznie do *training loss* - jest to suma błędów dla każdego przykładu w zbiorze walidacyjnym. Mierzona po każdej epoce informuje, czy model wymaga dalszego dostrojenia lub regulacji. Wyświetlana jest krzywa uczenia, aby była możliwa analiza, czy model działa poprawnie.

Na wykresach przedstawiających historie treningu wybranych modeli widzimy, że zarówno *validation loss* jak i *training loss* stopniowo zmniejszają się i stabilizują w określonym punkcie. Możemy zatem założyć, że modele są dobrze dobrane i nie uległy przeuczeniu.

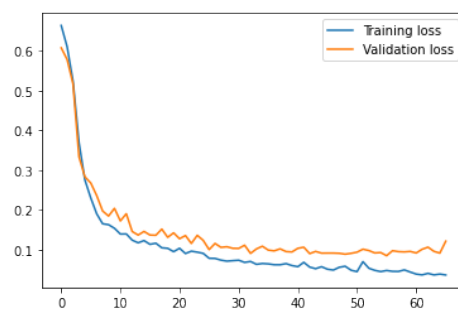
LSTM



Rysunek 3.1: Wizualizacja wybranego modelu LSTM dla całego zbioru testowego

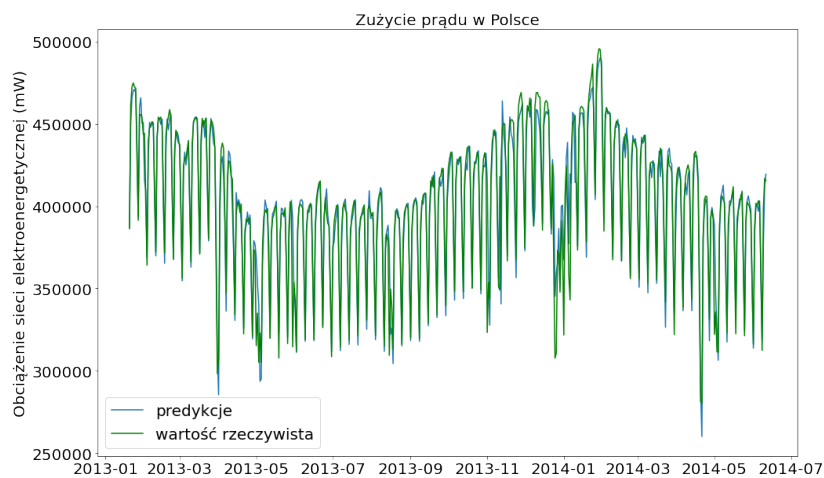


Rysunek 3.2: Wizualizacja wybranego modelu LSTM dla ostatnich 50 dni ze zbioru testowego

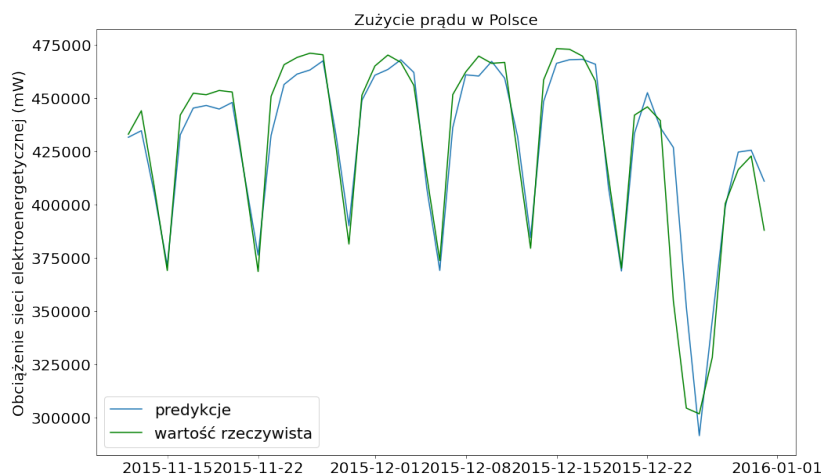


Rysunek 3.3: Historia treningu wybranego modelu LSTM

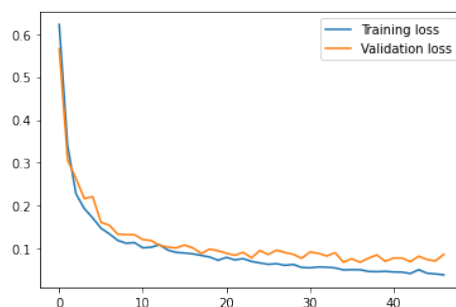
GRU



Rysunek 3.4: Wizualizacja wybranego modelu GRU dla całego zbioru testowego



Rysunek 3.5: Wizualizacja wybranego modelu GRU dla ostatnich 50 dni ze zbioru testowego



Rysunek 3.6: Historia treningu wybranego modelu GRU

3.1.1 Porównanie modeli

Ustawienia sieci dające najlepsze rezultaty dla poszczególnych państw zostały przedstawione w poniższych tabelach:

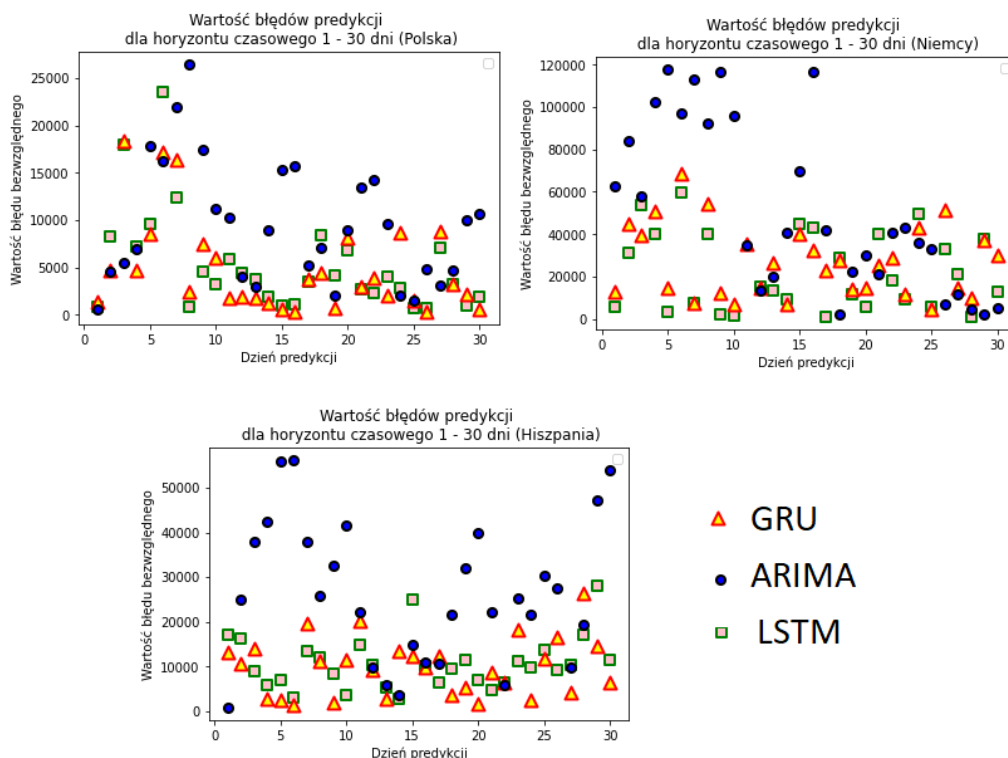
parametr	Polska	Hiszpania	Niemcy
ilość warstw sieci	5	2	5
długość sekwencji w oknie	21	28	21
ilość neuronów w najniższej warstwie	82	74	118

Tabela 3.1: Porównanie najlepszych ustawień sieci LSTM dla wybranych państw

parametr	Polska	Hiszpania	Niemcy
ilość warstw sieci	3	5	2
długość sekwencji w oknie	14	28	21
ilość neuronów w najniższej warstwie	70	10	118

Tabela 3.2: Porównanie najlepszych ustawień sieci GRU dla wybranych państw

W celu zobrazowania różnic dokładności prognoz poszczególnych modeli, obliczono jak zmieniają się wartości błędu $RMSE$ predykcji obciążenia sieci elektroenergetycznej dla kolejnych dni. Zostały one przedstawione na wykresie:



Rysunek 3.7: Błąd bezwzględny w zależności od horyzontu czasowego dla poszczególnych modeli

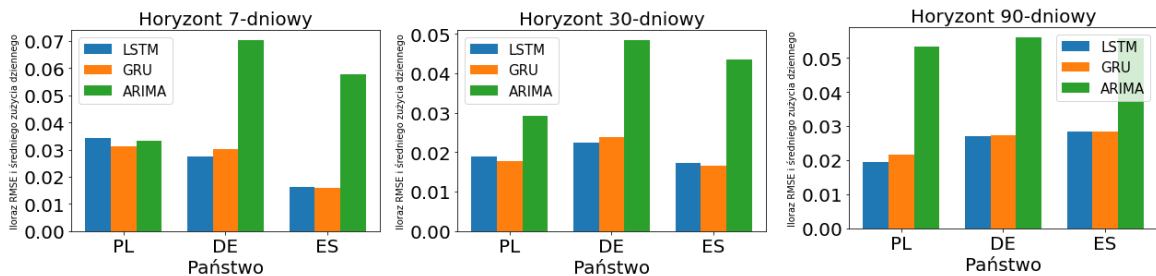
Wykres uwidacznia również, czy dokładność predykcji zmienia się w czasie: jeśli wartość błędu bezwzględnego w czasie zwiększa się - błędy w prognozach są większe, jeśli zmniejsza się - błędy także się zmniejszają.

Dokładne wartości wybranych metryk błędu $RMSE$, MAE , $MAPE$ oraz ME zostały również przedstawione w tabelach w paragrafach 3.3 - 3.7 odpowiadającym różnym horyzontom czasowym. W związku z początkiem rozpoczęcia predykcji w dniu 27 stycznia, model nie musi dostosowywać się do wahań obciążeń sieci elektroenergetycznej związanej ze zmianami długości dnia oraz temperatury, co jest ważne szczególnie przy krótkich horyzontach czasowych: 7-dniowym oraz 14-dniowym.

3.2 Porównanie dla różnych horyzontów czasowych

Jednym z celów badawczych pracy jest odpowiedzenie na pytanie, czy da się jednoznacznie wybrać jeden najlepszy model spośród testowanych powyżej. Poprawne określenie krótko- i długoterminowych zależności może być łatwiejsze do uzyskania w przypadku określonych modeli. W celu zobrazowania, jak zachowuje się błąd RMSE dla prostego modelu ARMA, sieci neuronowych wykorzystujących komórki LSTM

i GRU podczas prognozowania szeregów czasowych o odmiennych kształtach (lecz podobnych sezonowościach), został on przeskalowany. Na wykresach widzimy wartość błędu RMSE dla konkretnych horyzontów czasowych: 7-dniowego, 30-dniowego oraz 90-dniowego, która została podzielona przez średnią wartość dziennego obciążenia sieci elektroenergetycznej odpowiedniego państwa:



Rysunek 3.8: Przeskalowany błąd RMSE w zależności od horyzontu czasowego dla poszczególnych modeli

Zgodnie z przypuszczeniami, wyniki dla komórek GRU i LSTM są do siebie bardzo zbliżone. Nie jest to zaskakujące z uwagi na podobną metodę działania, mimo uproszczonej architektury komórki GRU. Z analizy wyników predykcji i wielkości błędów wynika, że proste modele statystyczne takie jak ARIMA radzą sobie jednak gorzej niż sieci neuronowe. Lepsze wyniki można byłoby uzyskać stosując podział na mniejsze szeregi czasowe eliminujące wahania sezonowe - w tym wypadku odpowiadające poszczególnym dniom tygodnia[22, 23].

3.3 Horyzont czasowy: 7 dni

Model	RMSE	MAE	MAPE	ME
Naiwny op. 1	38660.35	26191.57	6.29%	2510.71
Naiwny op. 7	62955.26	45302.57	3.74%	45302.57
LSTM	13332.56	11406.42	0.11%	-4442.56
GRU	12056.67	10163.38	0.14%	-4886.69
ARIMA	12872.09	10493.9	2.42%	-10493.9

3.4 Horyzont czasowy: 14 dni

Model	RMSE	MAE	MAPE	ME
Naiwny op. 1	37465.9	25214.07	5.95%	1846.07
Naiwny op. 7	45699.91	29486.57	3.17%	29486.57
LSTM	9817.75	7472.07	0.11%	-3332.84
GRU	8976.68	6699.85	0.12%	-2112.69
ARIMA	13356.07	11046.85	2.62%	-11046.85

3.5 Horyzont czasowy: 30 dni

Model	RMSE	MAE	MAPE	ME
Naiwny op. 1	38047.82	26584.77	6.27%	1291.03
Naiwny op. 7	33430.67	20462.97	4.65%	9016.03
LSTM	7302.38	5198.29	0.11%	-2573.03
GRU	6905.64	4864.38	0.12%	-1977.57
ARIMA	11384.24	9431.89	2.25%	-8810.38

3.6 Horyzont czasowy: 60 dni

Model	RMSE	MAE	MAPE	ME
Naiwny op. 1	335613.29	24594.97	5.87%	468.97
Naiwny op. 7	24448.89	13828.12	3.17%	2962.25
LSTM	6211.81	4637.3	0.11%	-2220.9
GRU	5799.25	4291.19	0.1%	-1758.26
ARIMA	15897.61	12243.63	2.88%	981.9

3.7 Horyzont czasowy: 90 dni

Model	RMSE	MAE	MAPE	ME
Naiwny op. 1	36453.35	25197.31	6.14 %	378.16
Naiwny op. 7	27661.53	15109.81	3.74 %	-500.12
LSTM	7522.78	5448.88	0.14%	-1329.9
GRU	8342.1	5779.05	0.12%	-423.92
ARIMA	20664.14	14840.58	3.69%	2192.18

W przypadku modelu ARIMA widać znaczący wzrost błędu RMSE wraz z długością horyzontu czasowego. Sieci neuronowe są dużo bardziej odporne na rozwijanie błędów w czasie.

3.8 Wpływ zmienności szeregu na wyniki predykcji

Jako generalną miarę zmienności szeregu czasowego w przypadku porównywania danych dotyczących zużycia energii elektrycznej różnych państw możemy przyjąć średni procentowy błąd *MAPE* naiwnego modelu, który jako prognozy przyjmuje wartości obciążeń sprzed 7 dni [24]: Poniżej znajdują się wartości współczynnika zmienności

Polska	Niemcy	Hiszpania
3.74 %	5.38%	5.73%

Tabela 3.3: Błąd MAPE modelu naiwnego dla 90-dni predykcji

szeregu w określonych dniach tygodnia. Wartości skrajne zostały zaznaczone kolorami -

wartość wskazująca na najmniejszą zmienność w ciągu tygodnia - kolorem zielonym, na największą - czerwonym.

CV	poniedziałek	wtorek	środa	czwartek	piątek	sobota	niedziela
PL	9,931	8,886	8,614	9,322	8,824	8,699	9,548
ES	8,398	7,895	7,486	8,121	8,496	7,349	7,957
DE	9,471	7,803	7,396	8,483	8,491	8,071	8,601

Rysunek 3.9: Współczynnik zmienności dla poszczególnych dni tygodnia

W poniższej tabeli zestawione zostały błędy *RMSE* osiągnięte przez modele sieci neuronowych, pogrupowane według dni tygodnia. Zgodnie z intuicją, sieci neuronowe zdecydowanie lepiej radzą sobie z przewidywaniami wartości dla dni tygodnia o mniejszej zmienności. Spowodowane jest to łatwiejszym przewidywaniem wartości, które mniej odbiegają od średniej. Nawet przy dłuższej sekwencji uczącej, w wypadku dużej zmienności dokładniejsza wartość może być trudna do oszacowania przez zbyt małą ilość możliwych do znalezienia wzorców i większe wahania przypadkowe.

	LSTM PL	GRU PL	GRU DE	LSTM DE	GRU ES	LSTM ES
day						
Czw	17542.895374	17629.322863	67395.173106	64856.700887	25152.539153	25021.347826
Niedz	8597.841376	9613.086144	36834.416788	29017.029195	19754.070558	15641.299822
Pon	18108.372655	15751.484784	68999.183109	51978.270441	26366.370967	23707.769659
Pt	14634.637621	14773.404983	60878.679280	63317.219447	23747.040114	24103.551386
Sb	11373.853552	10284.765818	29284.372269	40053.294452	12654.912370	12922.461259
Wt	13911.210022	14235.121223	41338.139424	35228.192077	21292.983086	19948.896649
Śr	12826.518645	14667.475993	46220.056697	39019.782856	23788.503824	24228.697030

Rysunek 3.10: RMSE dla poszczególnych dni tygodnia

3.9 Wpływ wybranej normalizacji na wyniki modelu

Zdecydowano się porównać 2 popularne sposoby na sprowadzanie danych do podobnego zakresu, aby zobaczyć, czy wybrana normalizacja ma znaczący wpływ na wyniki modelu. Pierwszy z nich, wspomniany w rozdziale "Budowa modeli", opisujemy wzorem:

$$z = \frac{x - \mu}{\sigma}, \quad \text{gdzie:} \quad (3.1)$$

- μ to średnia populacji,

- σ - odchylenie standardowe z populacji,
- x - wektor danych wejściowych.

Drugi sposób opiera się na wartościach skrajnych badanych danych i nazywany jest normalizacją min-max. Polega na zastosowaniu następującego wzoru na danych wejściowych:

$$z = \frac{x - \min(x)}{\max(x) - \min(x)}, \quad \text{gdzie:} \quad (3.2)$$

- x - wektor danych wejściowych
- $\min(x)$ - najmniejsza osiągnięta wartość x
- $\max(x)$ - największa osiągnięta wartość x

Przed uzyskaniem dokładnych predykcji możliwych do porównania ze zbiorem testowym, predykcje na danych znormalizowanych należy zdenormalizować poprzez odwrócenie kolejności przekształceń.

Powtórzone trening z drugim sposobem normalizacji przy takiej samej długości sekwencji uczącej, aby porównanie było miarodajne. Model jednak nadal dobierał parametry za pomocą optymalizatora. Można zauważyć, że dla takich samych długościach sekwencji uczących, różnią się ilości neuronów oraz warstw w modelach korzystających z danych komórek.

Poprzez zastosowanie innego podejścia do standaryzacji, udało się uzyskać także mniejsze wartości błędów. Poniżej znajduje się porównanie błędów *RMSE* dla sieci LSTM przewidującej wartości dla szeregu czasowego zużycia energii w Polsce:

standaryzacja \ horyzont	7	14	30	60	90
$\frac{x-\mu}{\sigma}$	13332.56	9817.75	7302.38	6211.81	7522.78
$\frac{x-\min(x)}{\max(x)-\min(x)}$	5956.66	5647.57	5856.75	6152.63	11454.86

Rozdział 4

Podsumowanie

W pracy zostały porównane 3 modele stosowane do predykcji kolejnych wartości szeregów czasowych: statystyczny model ARIMA oraz sieci neuronowe zbudowane z komórek GRU lub LSTM. Przetestowane zostały różne architektury modeli, aby za pomocą kryteriów informacyjnych i algorytmów optymalizacji osiągnąć jak najlepsze ustawienia. Oprócz standardowych miar błędów, porównane zostały również dokładności modeli naiwnych, które potwierdziły zasadność stworzonych modeli. Po analizie wyników dla 3 państw możemy stwierdzić, że przewidywania sieci neuronowych obarczone są dużo mniejszymi błędami pomiarowymi, jednak trening sieci neuronowej wiąże się także z większą ilością wykorzystanego czasu, mimo korzystania z optymalizatorów przerywających trening, gdy sieć przestaje robić postępy.

W celu polepszenia wyników predykcji, możliwe jest zastosowanie podejścia hybrydowego - połączenia dwóch lub więcej metod lub zastosowania przekształceń standardowych komórek sieci GRU lub LSTM[25]. Skuteczny jest również podział szeregów czasowych na podrzędne szeregi odpowiadające konkretnym godzinom lub dniom tygodnia oraz zastosowanie innych metod normalizacji danych.

Literatura

- [1] B.N. Oreshkin, G.Dudek, P.Pelka, E.Turkina, *N-BEATS neural network for mid-term electricity load forecasting*, Appl. Energy 2021, 293, 116918.
- [2] P.J. Brockwell, R.A. Davis. *Time series: Theory and Methods*, Springer – Verlag (1991)
- [3] W. S. McCulloch, W. H. Pitts *A Logical Calculus of Ideas Immanent in Nervous Activity*, Bulletin of Mathematical Biophysics, Vol. 5, pp. 115-133 (1943)
- [4] A. Géron, *Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 2nd ed., O'Reilly, 2019.
- [5] K. Benidis, S. S. Rangapuram, V. Flunkert, Y. Wang, D. Maddix, C. Turkmen, J. Gasthaus, M. Bohlke-Schneider, D. Salinas, L. Stella, F. Aubet, L. Callot, T. Januschowski *Deep Learning for Time Series Forecasting: Tutorial and Literature Survey*. arXiv:2004.10240 (2020)
- [6] S. Osowski, *Sieci neuronowe w ujęciu algorytmicznym*. Warszawa, 1996, Wydawnictwa Naukowo-Techniczne
- [7] H. Hewamalage, C. Bergmeir, K. Bandara *Recurrent Neural Networks for Time Series Forecasting: Current Status and Future Directions*. arXiv:1909.00590 (2019)
- [8] S. Hochreiter, J. Schmidhuber, *Long short-term memory*. *Neural Computation* **9**, 1735–1780 (1997)
- [9] G. Dudek, S. Smyl, P. Pełka *Recurrent Neural Networks for Forecasting Time Series with Multiple Seasonality: A Comparative Study*. arXiv:2203.09170 (2022)
- [10] F. Marmolejo-Ramos, T. Siva Tian, *The shifting boxplot. A boxplot based on essential summary statistics around the mean*. International Journal of Psychological Research, Vol.3 (2010)
- [11] M.HubertaE.Vandervierenb *Computational Statistics & Data Analysis*, Vol. 52, pp. 5186-5201 (2008)
- [12] Z. Bobowski *Wybrane metody statystyki opisowej i wnioskowania statystycznego*, Wydawnictwo WWSZiP, Wałbrzych, 2004

- [13] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, G. M. Ljung *Time Series Analysis: Forecasting and Control*, 5th Edition, Published by John Wiley and Sons Inc., Hoboken, New Jersey (2015)
- [14] J.T. Springenberg, A. Klein, S. Falkner, F. Hutter, *Bayesian Optimization with Robust Bayesian Neural Networks*, University of Freiburg, 2016 *Advances in neural information processing systems* 29
- [15] W. A. Fuller, *Introduction to Statistical Time Series*. New York: John Wiley and Sons (1976)
- [16] D.J. Lizotte, *Practical Bayesian Optimization*, University of Alberta, 2008
- [17] D. Lee, P. Schmidt, *On the power of the KPSS test of stationarity against fractionally-integrated alternatives*, Journal of Econometrics, Volume 73, Issue 1, 1996, Pages 285-302
- [18] D. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization*, Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015
- [19] H. Akaike, (1998). *Information Theory and an Extension of the Maximum Likelihood Principle*. W: E. Parzen, K. Tanabe, G. Kitagawa (red.) *Selected Papers of Hirotugu Akaike. Springer Series in Statistics*. Springer, New York, NY.
- [20] G. Schwarz *Estimating the Dimension of a Model*, The Annals of Statistics, Vol. 6, No. 2 (Mar., 1978), Published by: Institute of Mathematical Statistics
- [21] G. Dudek *STD: A Seasonal-Trend-Dispersion Decomposition of Time Series*. arXiv:2204.10398 (2022)
- [22] G. Dudek *Modele arima do krótkoterminowego prognozowania obciążeń systemów elektroenergetycznych* Rynek Energii Nr 2, pp. 94-98 (2012)
- [23] L. J. Soares, M. C. Medeiros, *Modeling and forecasting short-term electricity load: A comparison of methods with an application to Brazilian data*. International Journal of Forecasting 24, pp. 630-644, (2008)
- [24] G. Dudek *Modele wygładzania wykładniczego do krótkoterminowego prognozowania obciążeń systemów elektroenergetycznych* Rynek Energii Nr 3 (2013)
- [25] S. Smyl, G. Dudek, P. Pełka *ES-dRNN: A Hybrid Exponential Smoothing and Dilated Recurrent Neural Network Model for Short-Term Load Forecasting*. arXiv:2112.02663 (2021)