

Raport 2

Komputerowa analiza szeregów czasowych

Analiza danych przy pomocy modelu ARMA

Antczak Jakub, Curkowicz Kinga

06.02.2024

Spis treści

1	Wstęp	2
2	Przygotowanie do analizy	3
2.1	ACF i PACF dla surowych danych	3
2.2	Dekompozycja szeregu	4
2.3	Różnicowanie	5
3	Modelowanie danych przy pomocy ARMA	7
3.1	Estymacja parametrów modelu	8
3.2	Ocena dopasowania modelu	9
4	Weryfikacja założeń dotyczących szumu	10
4.1	Wartość średnia	10
4.2	Wariancja	11
4.3	Autokorelacja	11
4.4	Normalność residuów	12
4.4.1	Wizualizacja	12
4.4.2	Kurtoza, współczynnik skośności, reguła 3 sigm	13
4.4.3	Testy statystyczne	13
4.5	Interpretacja wyników	14
5	Podsumowanie	14

1 Wstęp

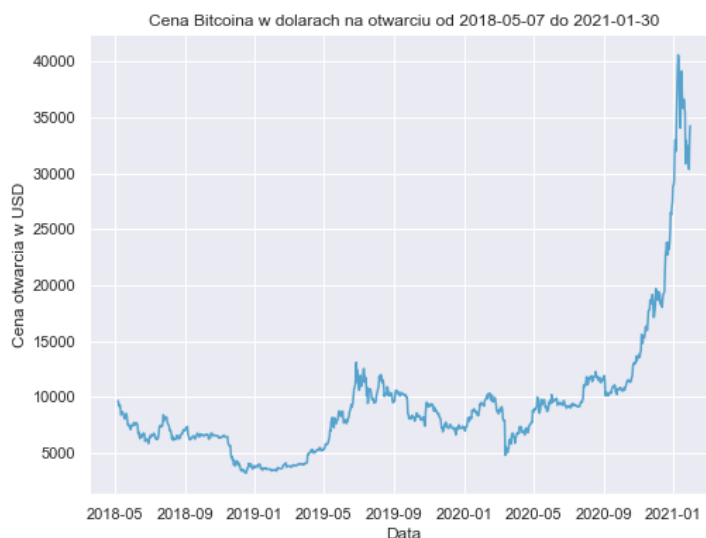
Niniejszy raport ma na celu analizę danych rzeczywistych przy pomocy modelu ARMA. Wybrane przez nas dane to cena Bitcoina w dolarach na otwarciu. Link do zbioru analizowanych danych można znaleźć na stronie Kaggle: <https://www.kaggle.com/datasets/ahmedadam415/digital-currency-time-series>.

Badany okres to 05.07.2018 – 30.01.2021, jednak ze względu na zamknięcie giełdy w weekendy i święta zbiór danych zawiera 1000 obserwacji.

Date	Open_USD	High_USD	Low_USD	Close_USD
2018-05-21	31979.548288	32235.850624	31147.072	31433.002496
2018-05-20	30878.880896	32287.1936	30617.89056	31979.585792
2018-05-19	30895.832704	31468.85632	30362.225792	30878.880896
2018-05-18	30148.790528	31027.846784	29672.78976	30902.583424
2018-05-17	31240.832	31743.3856	29924.4416	30158.691584
2018-05-16	31735.8848	31833.3952	30314.520704	31240.832
2018-05-15	32491.102848	33228.506496	31508.23552	31735.8848
2018-05-14	32552.384384	33303.514496	31042.0608	32490.990336
2018-05-13	31786.290176	32866.105344	31120.8192	32552.384384
2018-05-12	31525.637376	32429.258752	30577.0112	31750.661376
2018-05-11	33761.888384	33816.60672	31282.0864	31503.36
2018-05-10	34916.224	35235.458048	33641.088	33761.85088
2018-05-09	34421.1712	35216.256	33622.336	34916.224
2018-05-08	35122.496	35537.66528	33980.649216	34457.025024
2018-05-07	36232.689408	36340.138368	34432.4224	35122.496

Tabela 1: Bitcoin - pierwsze 15 obserwacji

Tabela 1 przedstawia udostępniane przez stronę informacje dla każdego dnia działania giełdy, czyli kolejno, cenę otwarcia, najwyższą i najniższą cenę akcji, cenę zamknięcia. Wybraną walutą jest dolar amerykański. W dalszej części raportu, skupimy się na cenie na otwarciu (**Open**) w zależności od dnia (**Date**).



Wykres 1: Ceny akcji na otwarciu w zależności od czasu

Jak można zobaczyć z wykresu 1, ceny akcji na otwarciu miały tendencję rosnącą.

2 Przygotowanie do analizy

2.1 ACF i PACF dla surowych danych

Aby zbadać czy autokorelacja rozpatrywanego szeregu czasowego wskazuje na jego stacjonarność w słabym sensie, często posługujemy się wykresami autokorelacji (ACF) i częściowej autokorelacji (PACF) dla danych. Estymator funkcji autokowariancji dany jest wzorem:

$$\hat{\gamma}(h) = \frac{1}{n} \sum_{t=1}^{n-|h|} (x_{t+|h|} - \bar{x})(x_t - \bar{x}),$$

gdzie \bar{x} to średnia próbkowa. Empiryczną funkcję autokorelacji definiujemy jako

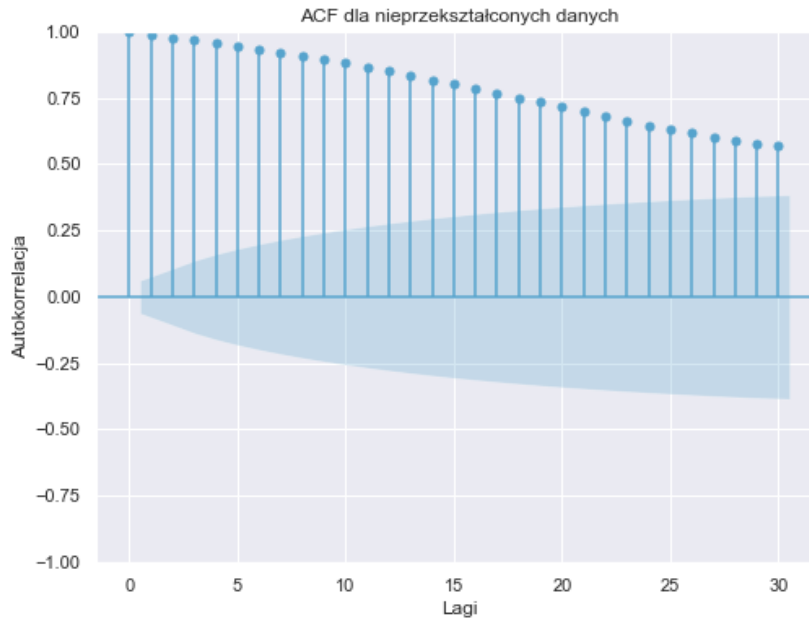
$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}.$$

Oczekiwany efekt to $\hat{\rho}(0) = 1$ i $\hat{\rho}(h) \approx 0$ dla $h \neq 0$. Empiryczna funkcja częściowej autokorelacji dana jest wzorem:

$$\hat{\phi}_h = \Gamma_h^{-1} \hat{\gamma}_h,$$

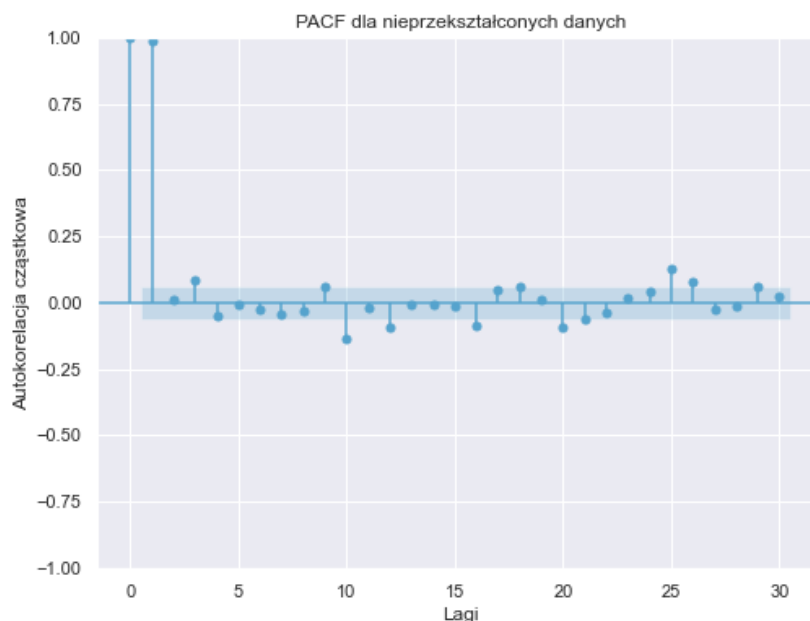
gdzie $\hat{\gamma}_h = [\hat{\gamma}(0), \dots, \hat{\gamma}(h)]$ i $\Gamma_h = [\hat{\gamma}(i-j)]_{i,j=1}^h$. W przypadku szeregu stacjonarnego w słabym sensie również oczekujemy, że dla względnie dużych h wartość $\hat{\phi}_h$ będzie w okolicach zera.

Stworzyliśmy wykres autokorelacji ACF 7 i częściowej autokorelacji PACF 9 dla surowych danych.



Wykres 2: ACF dla danych surowych

Wykres ACF 9 ukazuje, że istnieją znaczące wartości autokorelacji dla różnych opóźnień, co wskazuje na brak stacjonarności w badanym szeregu czasowym.



Wykres 3: PACF dla danych surowych

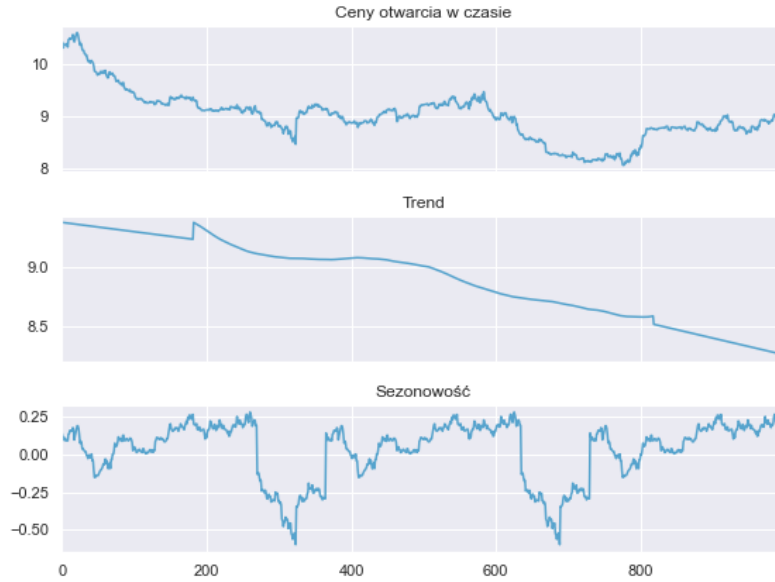
Przygotowanie danych do analizy zaczniemy od dekompozycji szeregu czasowego.

2.2 Dekompozycja szeregu

Analiza szeregów czasowych przy użyciu modelu ARMA wymaga, aby badany szereg był stacjonarny. Stacjonarność oznacza, że statystyczne właściwości szeregu czasowego, takie jak średnia i wariancja, są stałe w czasie, co ułatwia prognozowanie przy użyciu modeli czasowych.

Aby przeprowadzić analizę ARMA, często konieczne jest przekształcenie szeregu czasowego w postać stacjonarną. Wykorzystamy do tego dekompozycję szeregu czasowego, co oznacza usunięcie liniowego trendu oraz sezonowości.

Do naszych działań wykorzystaliśmy funkcję `seasonal_decompose` z pakietu `statsmodels`.

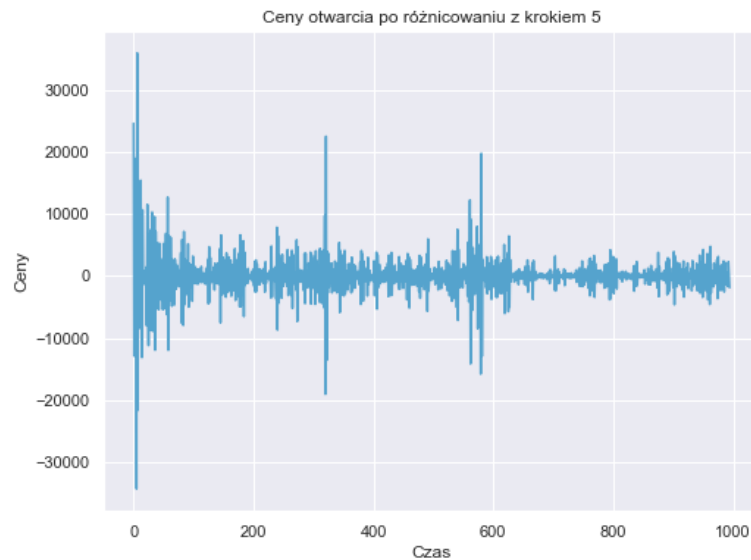


Wykres 4: Dane rzeczywiste, trend liniowy, sezonowość

2.3 Różnicowanie

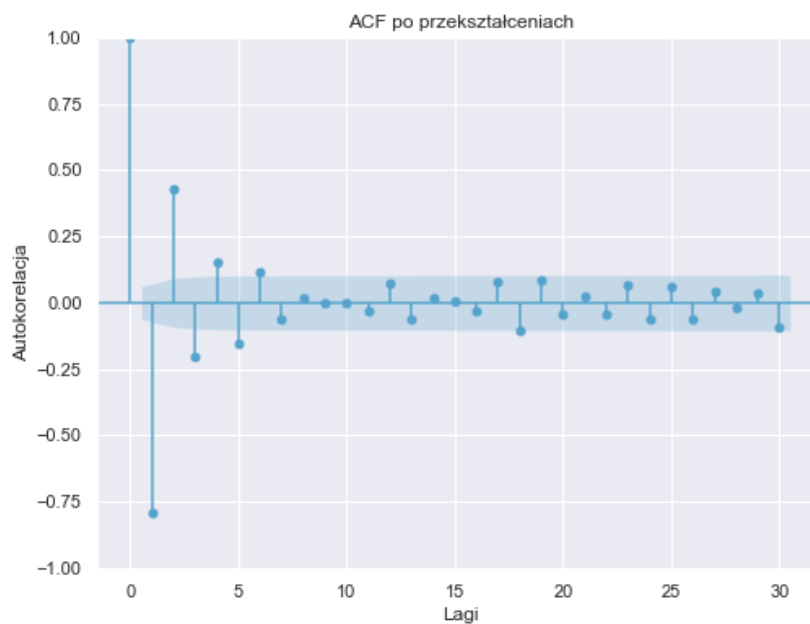
Różnicowanie z krokiem d to wykonanie następującej operacji na szeregu X_t . Wybór odpowiedniego kroku różnicowania d zależy od stopnia niestacjonarności w badanym szeregu. W praktyce, proces ten może być powtarzany wielokrotnie, aż do uzyskania stacjonarności, co sprawia, że szereg staje się bardziej odpowiedni do zastosowania modelu ARMA w analizie czasowej i prognozowaniu. Jest to ostatni krok, który podjęliśmy w celu doprowadzenia naszego szeregu do postaci stacjonarnej.

$$\tilde{X}_t = X_t - X_{t-d} \quad (1)$$

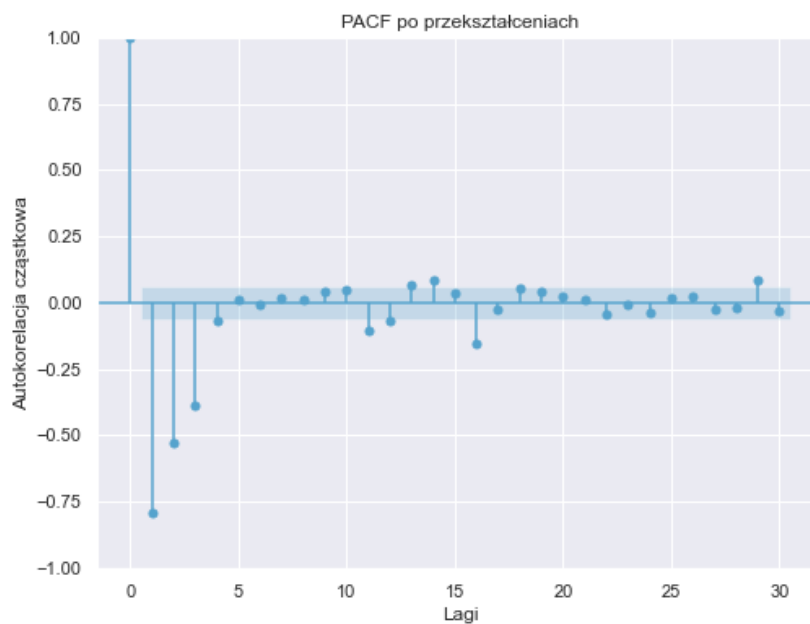


Wykres 5: Ceny po różnicowaniu

Na wykresie 5 prezentujemy ceny akcji po zróżnicowaniu. Przejdziemy teraz do ponownej wizualizacji ACF i PACF, tym razem już dla szeregu stacjonarnego.



Wykres 6: ACF po zróżnicowaniu



Wykres 7: PACF po zróżnicowaniu

Możemy zauważyć, że po zastosowaniu różnicowania szeregu danych, zarówno autokorelacja (ACF) jak i częściowa autokorelacja (PACF) szybciej zbiegają do zera wraz ze wzrostem opóźnienia. Ten szybszy spadek sugeruje, że oczyściliśmy szereg czasowy z niestacjonarności, eliminując w znacznym stopniu strukturalne zależności czasowe, które były obecne w pierwotnych danych.

Przeprowadziliśmy test Augmented Dickey-Fuller w celu spadania stacjonarności zróżnicowanych danych. Otrzymane wyniki prezentujemy w tabeli 2.

Parametr	Wartość
Statystyka ADF	-18.87085406368964
Wartość p	0.0
Wartości krytyczne (1%)	-3.4370955284183404
Wartości krytyczne (5%)	-2.8645180847192755
Wartości krytyczne (10%)	-2.568355689215736

Tabela 2: Wyniki testu ADF

Wyniki testu Augmented Dickey-Fuller zaprezentowane w tabeli 2 wskazują na silne dowody na stacjonarność analizowanego szeregu czasowego. Statystyka ADF wynosi -18.87 , co jest znacznie niższe od wartości krytycznych dla wszystkich poziomów istotności. Wartość p jest równa 0.0 , co potwierdza wysoką istotność statystyczną testu. To oznacza, że możemy odrzucić hipotezę zerową o niestacjonarności danych. Wartości krytyczne również potwierdzają ten wniosek, będąc wyraźnie niższe od statystyki ADF. W rezultacie, analizowany szereg czasowy może być uznany za stacjonarny zgodnie z wynikami testu ADF.

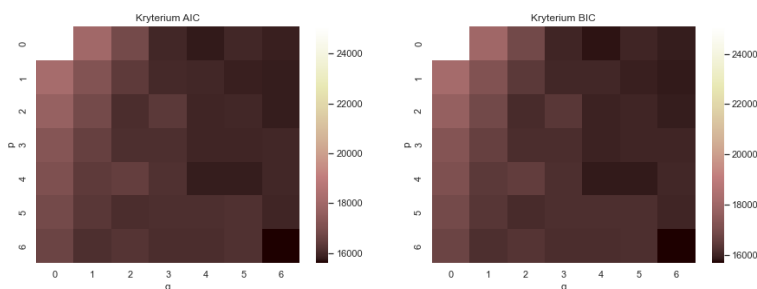
3 Modelowanie danych przy pomocy ARMA

Przejdziemy teraz do modelu $\text{ARMA}(p, q)$. Jest to autoregresyjno-średniowy model ruchomy, odnoszący się do słabo stacjonarnego szeregu czasowego $\{X_t\}_{t \in \mathbb{Z}}$, który spełnia równanie:

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

gdzie $\{Z_t\}_{t \in \mathbb{Z}}$ to biały szum o rozkładzie normalnym $WN(0, \sigma^2)$, a wielomiany $\phi(z) = 1 - \phi_1 z - \dots - \phi_p z^p$ oraz $\theta(z) = 1 + \theta_1 z + \dots + \theta_q z^q$ nie posiadają wspólnych pierwiastków. W modelu $\text{ARMA}(p, q)$, parametr p oznacza rząd modelu autoregresyjnego (AR), natomiast parametr q to rząd modelu średniej ruchomej MA.

Do dobrania rzędu modelu, wykorzystano kryterium informacyjne Akaikego (AIC), a w celu weryfikacji rzędu modelu, użyliśmy również Bayesowskie kryterium informacyjne Schwarza (BIC).



Wykres 8: Kryterium AIC ORAZ BIC

Dobrene przez nas parametry to p równe 6 oraz q równe 6. Skorzystaliśmy z funkcji `ARIMA` z pakietu `statsmodels.tsa.arima.model` w języku Python.

3.1 Estymacja parametrów modelu

Estymacja parametrów modelu ARMA(6,6) została przeprowadzona przy użyciu metody największej wiarygodności (MLE). Metoda ta polega na znalezieniu takich wartości parametrów autoregresji $(\phi_1, \phi_2, \dots, \phi_6)$ oraz średniej ruchomej $(\theta_1, \theta_2, \dots, \theta_6)$, które maksymalizują funkcję wiarygodności dla dostępnych danych. Funkcja wiarygodności opisuje prawdopodobieństwo uzyskania obserwowanych danych przy założeniu modelu ARMA(6,6).

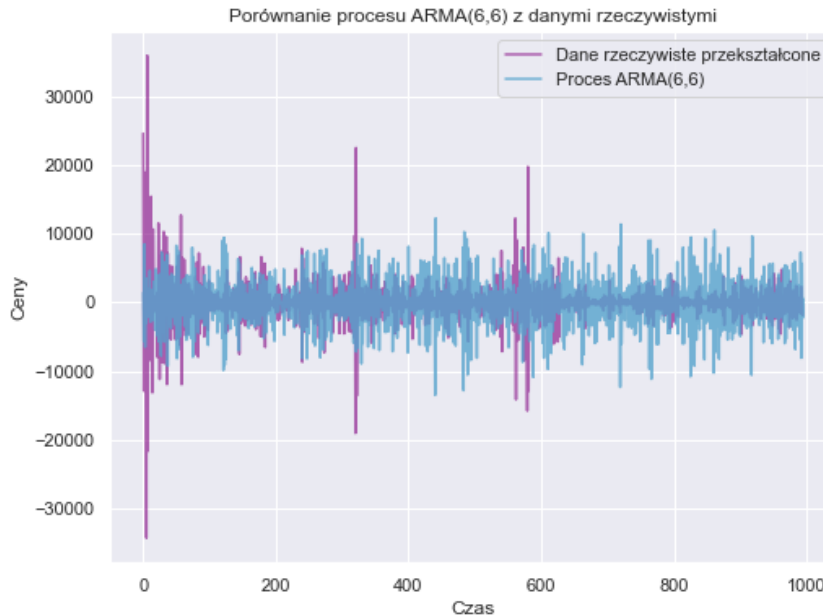
Model ARMA(6,6) możemy zapisać jako:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \phi_3 X_{t-3} + \phi_4 X_{t-4} + \phi_5 X_{t-5} + \phi_6 X_{t-6} + Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \theta_3 Z_{t-3} + \theta_4 Z_{t-4} + \theta_5 Z_{t-5} + \theta_6 Z_{t-6}$$

gdzie:

- X_t to wartość szeregu czasowego w chwili t ,
- Z_t to biały szum o rozkładzie normalnym $N(0, \sigma^2)$,
- $\phi_1, \phi_2, \dots, \phi_6$ to parametry autoregresji,
- $\theta_1, \theta_2, \dots, \theta_6$ to parametry średniej ruchomej.

Proces estymacji polega na znalezieniu wartości tych parametrów, które najlepiej opisują obserwowane dane, maksymalizując funkcję wiarygodności.



Wykres 9: Porównanie procesu ARMA(6,6) z danymi rzeczywistymi

Poniżej znajduje się tabela przedstawiająca wyniki estymacji parametrów modelu ARMA(6,6):

Parametr: Numer parametru w modelu ARMA.

Wartość: Wartość estymowanego parametru.

Błąd standardowy: Błąd standardowy estymowanego parametru.

Parametr	Wartość	Błąd standardowy	Wartość p
AR(1)	-1.4999	0.090	< 0.001
AR(2)	0.0991	0.063	0.113
AR(3)	1.0470	0.065	< 0.001
AR(4)	0.6706	0.051	< 0.001
AR(5)	0.4647	0.050	< 0.001
AR(6)	0.1727	0.028	< 0.001
MA(1)	-1.9785	0.244	< 0.001
MA(2)	-0.7797	0.557	0.161
MA(3)	3.4868	0.205	< 0.001
MA(4)	-0.7313	0.484	0.131
MA(5)	-1.9771	0.402	< 0.001
MA(6)	0.9826	0.085	< 0.001

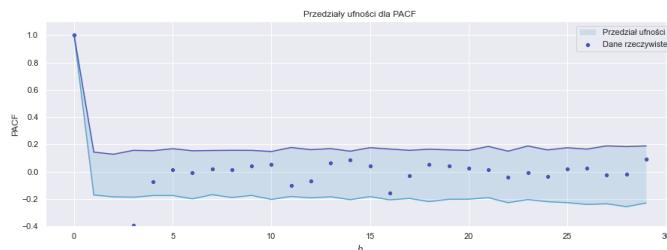
Tabela 3: Wyniki estymacji parametrów modelu ARMA(6,6)

Wartość p: Wartość p testu statystycznego dla danego parametru. Wartości mniejsze niż 0.05 są uznawane za istotne statystycznie.

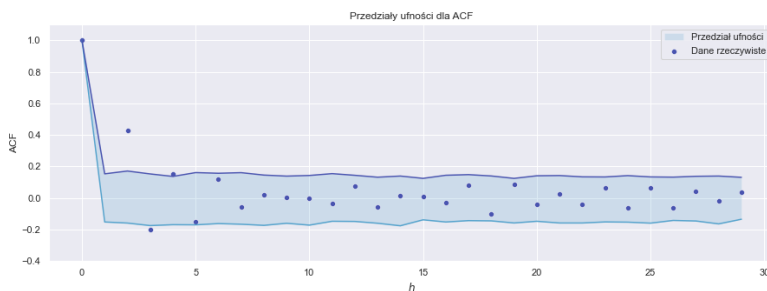
Wartości estymatorów ϕ_t są przedstawione w tabeli 3 jako $ar.L(t)$, a wartości estymatorów θ_t jako $ma(t)$.

3.2 Ocena dopasowania modelu

W kolejnej części sprawdziliśmy dopasowanie naszego modelu ARMA(6,6). W tym celu generowaliśmy szereg ARMA(6,6) wielokrotnie i za każdym razem wyznaczaliśmy autokorelację dla h od 0 do 30 oraz częściową autokorelację dla h od 1 do 30. Następnie dla każdego h policzyliśmy kwantyle rzędu $\alpha/2$ oraz $1 - \alpha/2$ dla $\alpha = 0.05$. W ten sposób otrzymano przedziały ufności dla ACF oraz PACF. Żeby ocenić dopasowanie modelu do danych sprawdzono, czy ich autokorelacja oraz częściowa autokorelacja mieszczą się w wyznaczonych przedziałach. Prezentujemy to na wykresie 10 oraz 11.



Wykres 10: Przedziały ufności dla PACF



Wykres 11: Przedziały ufności dla ACF

Własność ta zachodzi dla każdego h zarówno w przypadku ACF, jak i PACF, tak więc można wnioskować o poprawności dobranego modelu ARMA.

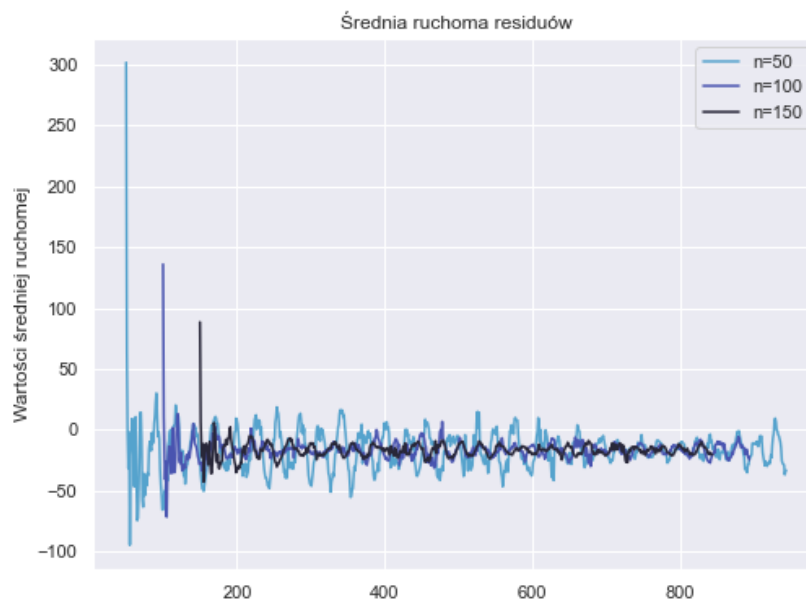
4 Weryfikacja założeń dotyczących szumu

W tym punkcie skupimy się na analizie residuów. Założenia, które muszą spełniać residua, to:

1. Czy ich wartość oczekiwana wynosi 0?
2. Czy mają stałą wariancję?
3. Czy są nieskorelowane?
4. Czy pochodzą z rozkładu normalnego?

4.1 Wartość średnia

Zacznijmy od sprawdzenia, czy wartość średnia residuów jest równa 0.



Wykres 12: Średnia ruchoma

Wyliczona wartość średnia z residuów wynosi 12.86. Jest to wartość bardzo daleka od zera. Narysowano również wykres średniej ruchomej Wykres 12, która liczy średnią arytmetyczną dla n ostatnich okresów. Aby go narysować, najpierw obliczyliśmy średnią ruchomą zgodnie ze wzorem:

$$MA = \frac{1}{2p+1} \sum_{k=i-p}^{i+p} x_k, \quad (2)$$

gdzie p to n ostatnich okresów. W naszym przypadku $n = 50, 100, 150$.

Na wykresie 12 widać, że średnia arytmetyczna z n ostatnich obserwacji, dla każdego z analizowanych dni, nie układa się symetrycznie względem zera. Wraz ze wzrostem długości n , wykres wypłaszcza się. Prowadzi nas to do postawienia hipotezy, że residua nie mają wartości średniej równej 0.

W celu potwierdzenia, że wartość średnia nie wynosi 0, przeprowadźmy test t-Studenta na poziomie istotności $\alpha = 0.05$. Weźmy:

H_0 : wartość średnia residuów jest równa 0,

H_1 : wartość średnia residuów nie jest równa 0.

Do przeprowadzenia testu użyjemy funkcji `ttest_1samp` z biblioteki `scipy.stats` w Pythonie. W ten sposób uzyskaliśmy p-wartość równą około 0.71. Jest to znacznie więcej niż ustalone $\alpha = 0.05$. Funkcja zwraca również t-statystykę, czyli stosunek odchylenia szacowanej wartości parametru od jego hipotetycznej wartości do jego błędu standardowego. W tym przypadku jest ona równa 0.38. Otrzymane wyniki wskazują na to, że nie ma najmniejszych podstaw do odrzucenia hipotezy zerowej, jednak z wykresu 12, że średnia residuów nie zbiega do zera. Wartość statystyki testowej informuje nas o dużym błędzie, dlatego decydujemy się odrzucić hipotezę zerową i przyjąć alternatywną.

4.2 Wariancja

W kolejnym kroku przeprowadzanej analizy, sprawdzimy, czy badane residua mają stałą wariancję. Przeprowadźmy zmodyfikowany test Levene'a, który sprawdza, czy wszystkie podprzedziały mają stałą wariancję. Ustalmy poziom istotności $\alpha = 0.05$ oraz przyjmijmy:

H_0 : wariancja jest stała,

H_1 : wariancja nie jest stała.

W celu przeprowadzenia testu użyjemy funkcji `stats.levene` z biblioteki `stats` w Pythonie. Aby to zrobić podzieliliśmy nasze residua na cztery podgrupy:

- Grupa 1 zawiera residua, które są mniejsze od pierwszego kwantyla (25. percentyla).
- Grupa 2 zawiera residua, które są większe lub równe pierwszemu kwantylowi (25. percentylowi) i mniejsze od mediany.
- Grupa 3 zawiera residua, które są większe lub równe medianie i mniejsze od trzeciego kwantyla (75. percentyla).
- Grupa 4 zawiera residua, które są większe lub równe trzeciemu kwantylowi (75. percentylowi).

Otrzymaliśmy p-wartość równą $3.21 \cdot 10^{-7}$. Jest to znacznie mniejsze od poziomu istotności α , dlatego też jesteśmy zmuszeni odrzucić hipotezę zerową i przyjąć hipotezę alternatywną.

4.3 Autokorelacja

Kolejnym krokiem analizy jest sprawdzenie, czy residua są od siebie niezależne. Użyjemy funkcji `plot_acf` z biblioteki `statsmodels` w Pythonie, aby narysować funkcję autokorelacji dla residuów. Zdefiniujemy funkcję autokorelacji.

Funkcja autokorelacji (ρ) przy przesunięciu h , gdzie h przyjmuje wartości od $-n$ do n (gdzie n to liczba residuów), mierzy stopień korelacji między wartościami residuów oddalonymi o h przedziałów od siebie w szeregach czasowych. W przypadku teoretycznej funkcji autokorelacji (ρ), dla $h = 0$, wartość wynosi 1, co oznacza doskonałą korelację z samym sobą. Dla $h \neq 0$, wartość wynosi 0, co oznacza brak korelacji.

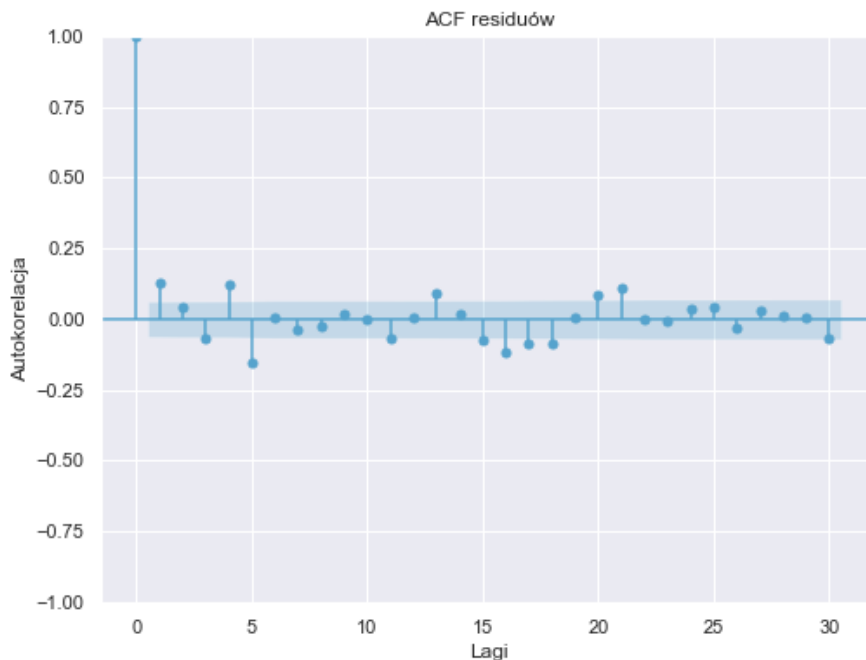
Będziemy porównywać ją z jej empirycznym odpowiednikiem:

$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)} \quad (21)$$

gdzie $\hat{\gamma}$ to empiryczna funkcja autokowariancji dana wzorem:

$$\hat{\gamma}(h) = \frac{1}{n} \sum_{i=1}^{n-|h|} (e_{i+|h|} - \bar{e})(e_i - \bar{e}) \quad (22)$$

Weźmy wektor $h = 0 : 1 : 30$. Sprawdźmy autokorelację pomiędzy każdymi h -tymi residuami. Wyniki przedstawiony został na Wykresie 13. Wykres ten zdaje się potwierdzać tezę, że residua są nieskorelowane. W zerze funkcja autokorelacji wynosi 1, a w pozostałych punktach jest w okolicach zera, przy czym większość "słupków" nie wychodzi znacznie poza zaznaczony przedział ufności.



Wykres 13: Autokorelacja residuów

Sam wykres nie wystarczy do stwierdzenia czy residua są nieskorelowane. Przeprowadźmy test Ljung-Boxa, który służy do sprawdzenia czy dane są nieskorelowane. Wykorzystajmy funkcję `accor_ljungbox` z pakietu `statsmodels`. Ustalmy nasze hipotezy:

$$H_0 : \text{residua są nieskorelowane,}$$

$$H_1 : \text{residua nie są nieskorelowane.}$$

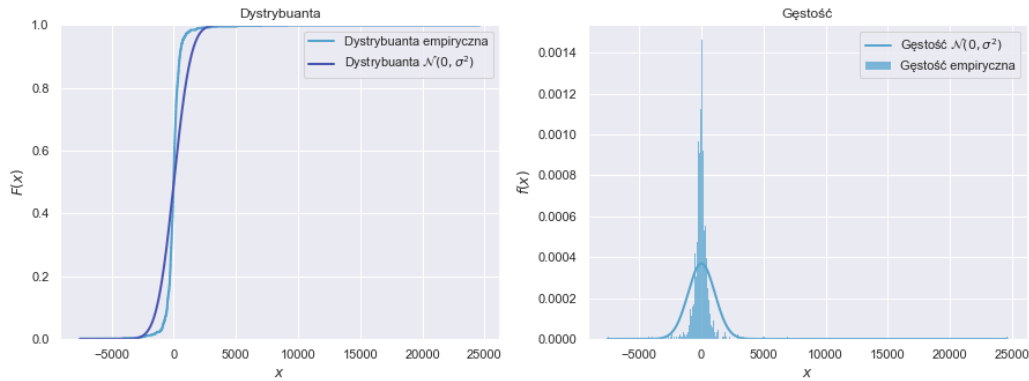
Po przeprowadzeniu testu, p-wartości osiągają wartości rzędów od 10^{-6} do 10^{-16} . Większość p-wartości jest o wiele mniejsza niż o poziom istotności α , przez co jesteśmy zmuszeni odrzucić hipotezę zerową i przyjąć, że residua nie są nieskorelowane.

4.4 Normalność residuów

Ostatnim krokiem naszej analizy będzie sprawdzenie, czy residua pochodzą z rozkładu normalnego.

4.4.1 Wizualizacja

Zacznijmy od porównania dystrybuanty i gęstości empirycznej residuów z teoretycznym rozkładem normalnym $\mathcal{N}(0, S^2)$.



Wykres 14: Porównanie dystrybuanty i gęstości empirycznej z teoretycznymi

Jak widzimy na wykresie 14 dystrybuanty i gęstości nie pokrywają się.

Zauważmy, że w okolicach 0 na wykresie gęstości występuje skok, który mówi nam o tym, że nasze błędy są często blisko 0, jednak jak pokazaliśmy wyżej ich średnia wcale nie jest bliska zera.

4.4.2 Kurtoza, współczynnik skośności, reguła 3 sigm

Wzór na kurtozę i skośność można znaleźć w sekcji ??, natomiast w celu ich obliczenia użyliśmy funkcji wbudowanych `skew` oraz `kurtosis` z pakietu `scipy.stats`. Uzyskaliśmy, że kurtoza residuów jest równa 278.39. Dla rozkładu normalnego powinna ona wynosić 3. Te liczby nie są zbliżone do siebie, czego można było się spodziewać po wykresie 14.

W rozkładzie normalnym, przy $\mu = 0$ współczynnik skośności wynosi 0. Współczynnik skośności, w naszym przypadku wynosi 12.46. Jest to wartość daleka od pożądanej.

Sprawdźmy teraz regułę trzech sigm. Wiemy, że dla zmiennej losowej $X \sim \mathcal{N}(\mu, \sigma^2)$ zachodzi reguła trzech sigm:

$$P(\mu - 3\sigma \leq X \leq \mu + 3\sigma) \approx 0.99 \quad (3)$$

Sprawdźmy czy nasze dane spełniają równanie 3.

$$ecdf(3S) - ecdf(-3S) \approx 0.99 \quad (4)$$

`ecdf` to empiryczna funkcja dystrybuanty wyliczana za pomocą funkcji `stats.norm.cdf` z pakietu `scipy.stats`. Otrzymana wartość wynosi 1, co nie powinno nas dziwić bo jak widać na wykresie 14 gęstość jest mocno skoncentrowana wokół zera.

4.4.3 Testy statystyczne

W celu ostatecznego sprawdzenia czy residua pochodzą z rozkładu normalnego, przeprowadziliśmy dwa testy: test Kołmogorowa-Smirnova oraz test Shapiro-Wilka. Ustalmy poziomi istotności $\alpha = 0.05$ hipotezę zerową i alternatywną:

$$H_0 : \text{residua są z rozkładu normalnego,}$$

$$H_1 : \text{residua nie są z rozkładu normalnego.}$$

Do przeprowadzenia testów użyliśmy funkcji `kstest` oraz `shapiro` z biblioteki `scipy.stats`. Dla ks-testu p-wartość jest równa $9.03 \cdot 10^{-244}$, a dla Shapiro-Wilka 0.00. Obie te wartości są za małe. Jesteśmy zmuszeni odrzucić hipotezę zerową i przyjąć alternatywną.

4.5 Interpretacja wyników

Po przeanalizowaniu residuów wysnuliśmy wniosek, że nie pochodzą one z rozkładu normalnego. Wszystkie pozostałe założenia również nie zostały spełnione. Ważne, żeby obserwacje wizualne potwierdzić obliczeniami czy też testami statystycznymi. Po wykresie 13 moglibyśmy stwierdzić, że residua są niezależne, natomiast dopiero test statystyczny pokazał nam, że residua nie są nieskorelowane. Możemy stwierdzić, że wybrany model nie jest w pełni poprawny, mimo tego, że dane pokrywają się z ARMA(6,6).

5 Podsumowanie

Celem raportu była analiza danych dotycząca cen Bitcoina w USD na otwarciu przy pomocy modelu ARMA. Początkowo doprowadziliśmy nasz szereg do postaci stacjonarnej. Zastosowaliśmy takie metody jak dekompozycja oraz różnicowanie. Transformacja ta przyniosła oczekiwane efekty.

Do tak przygotowanych danych dobrano rząd modelu korzystając z kryteriów informacyjnych i otrzymano szereg ARMA(6,6), a następnie za pomocą funkcji wbudowanych wyestymowano wszystkie parametry modelu. W celu oceny dopasowania modelu porównano jego wykres z wykresem cen otwarcia doprowadzonych do postaci stacjonarnej oraz sprawdzono czy autokorelacja i częściowa autokorelacja mieszczą się w wyznaczonych przedziałach ufności. Ocena ta w obu przypadkach była pozytywna, w związku z czym stwierdzono, że model ARMA został dobrany prawidłowo.

Ostatnim etapem pracy była weryfikacja założeń dotyczących szumu. Sprawdzono, że residua nie układają się równomiernie względem zera oraz, że wariancja nie jest stała. Następnie używając testów statystycznych, stwierdzono, że residua nie są nieskorelowane. Dodatkowo porównano rozkład residuów ze standardowym rozkładem normalny, wykorzystując do tego wykres gęstości oraz dystrybuanty. Cała analiza residuów, powiedziała nam, że żadne założenie nie zostało spełnione.

Po podsumowaniu wszystkich aspektów, można stwierdzić, że model nie jest do końca poprawny, być może można by było zaproponować inny, taki jak ARIMA.