

# Raport 2

## Analiza szeregów czasowych

Zuzanna Kramarz, Isja Utemisov  
albumy 268836, 268889

11 grudnia 2023

### Spis treści

<b>1</b>	<b>Zadanie 2 - Dekompozycja szeregów czasowych - eliminacja trendu i sezonowości</b>	<b>3</b>
1.1	Cel zadania	3
1.2	Wybór Szeregu Czasowego	3
1.3	Analiza podstawowych własności szeregu	3
1.3.1	Wykres zwykły	3
1.3.2	Wykres sezonowy	4
1.3.3	Wykres miesięczny	5
1.3.4	Wykres ACF i PACF	6
1.3.5	Wykres pudełkowy	7
1.3.6	Wykres rozrzutu dla opóźnienia $h=1, \dots, 12$	8
1.4	Zastosowanie i porównanie metod dekompozycji	9
1.4.1	Dekompozycja na podstawie ruchomej średniej (funkcja <code>decompose()</code> )	9
1.4.2	Dekompozycja na podstawie modelu regresji (funkcja <code>tslm()</code> )	15
1.4.3	Dekompozycja STL oparta na metodzie loess (funkcja <code>stl()</code> )	22
1.4.4	Porównanie jakości dopasowania poszczególnych modeli dekompozycji	33
1.5	Zastosowanie transformacji Boxa–Coxa	34
1.5.1	Dekompozycja na podstawie ruchomej średniej	34
1.5.2	Dekompozycja na podstawie modelu regresji	35
1.5.3	Dekompozycja STL oparta na metodzie loess	40
1.6	Porównanie wyników eliminacji trendu i sezonowości	41
<b>2</b>	<b>Zadanie 3 - Zastosowanie wskaźników analizy technicznej w analizie szeregów czasowych</b>	<b>43</b>
2.1	Opis wskaźników	43
2.1.1	MACD	43
2.1.2	STS	44
2.1.3	SMI	44
2.1.4	Wstęga Bollingera	45
2.1.5	RSI	45
2.1.6	SMA	45
2.2	Wizualizacja wybranych wskaźników	45
2.2.1	MACD	52
2.2.2	SMI	55
2.2.3	Wstęga Bollingera	57
2.2.4	RSI	61
2.2.5	SMA	66
2.3	Wpływ wyboru parametrów dla poszczególnych wskaźników na otrzymane wnioski	73
2.3.1	MACD	73
2.3.2	SMI	75

2.3.3	Wstęga Bollingera . . . . .	78
2.3.4	RSI . . . . .	81
2.3.5	SMA . . . . .	87

# 1 Zadanie 2 - Dekompozycja szeregów czasowych - eliminacja trendu i sezonowości

## 1.1 Cel zadania

Celem zadania jest przeprowadzenie analizy szeregu czasowego w kontekście eliminacji trendu i sezonowości. Analiza obejmuje zastosowanie oraz porównanie trzech metod dekompozycji: `decompose()`, `tslm()`, oraz `stl()`. W trakcie analizy uwzględnione zostaną także różne parametry i transformacje danych w celu oceny wpływu na jakość dopasowania.

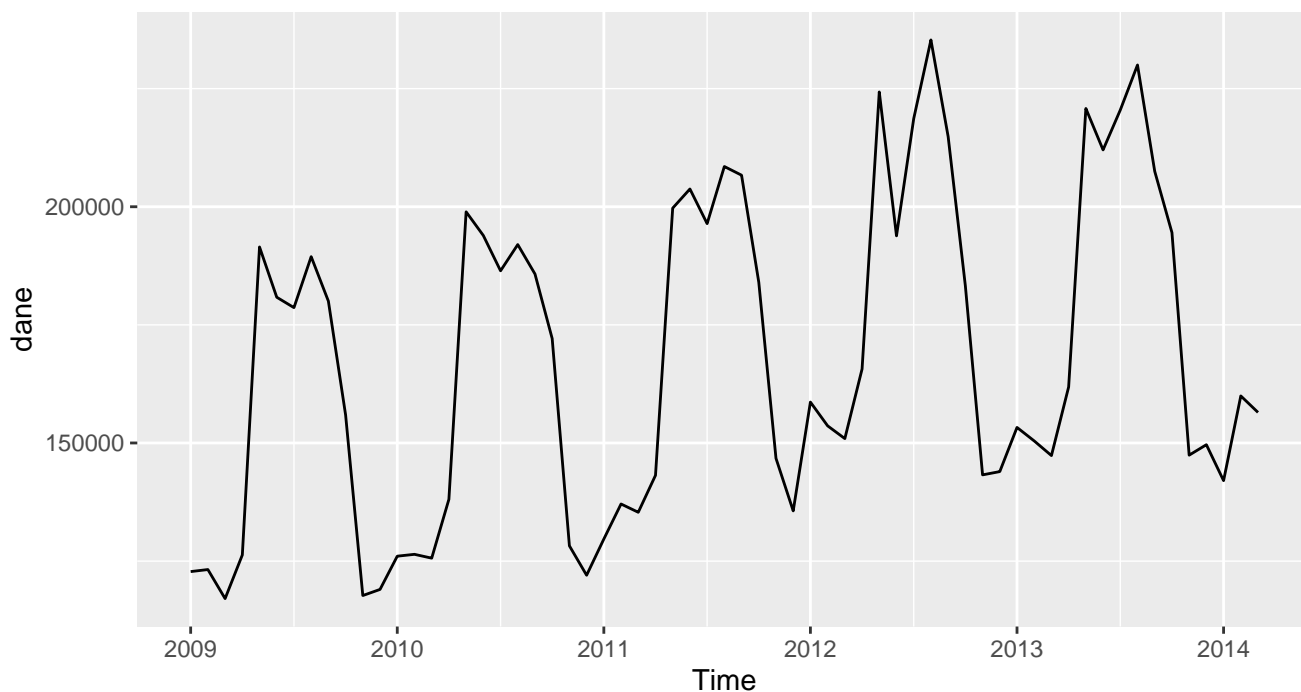
## 1.2 Wybór Szeregu Czasowego

Analizę przeprowadzimy dla szeregu czasowego *hotele{TSAFBook}*, pakiet danych "hotele" w ramach zbioru TSA-FBook zawiera informacje dotyczące miesięcznej liczby turystów (gości hotelowych) odwiedzających województwo Dolnośląskie w Polsce.

## 1.3 Analiza podstawowych własności szeregu

Dokonamy analizy wstępnej szeregu czasowego. Wykorzystamy do tego metody graficzne, takie jak wykresy zwykłe i sezonowe, aby zidentyfikować obecności trendu, sezonowości oraz obserwacji odstających.

### 1.3.1 Wykres zwykły



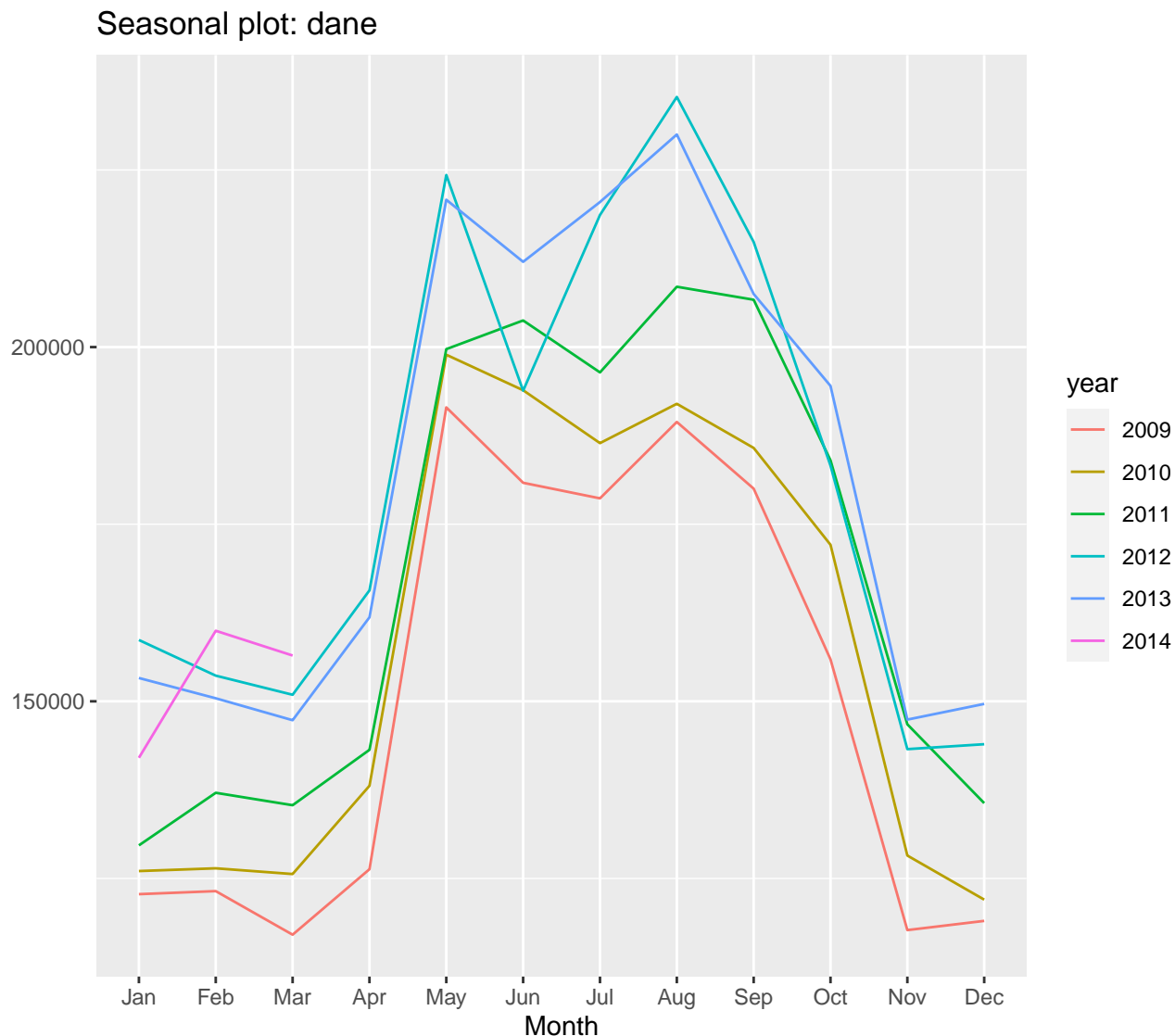
Rysunek 1: Wykresy zwykły dla danych szeregu czasowego - Hotele

Wnioski:

Obecność regularnych wzorców wzrostu i spadku, powtarzających się w określonych okresach, sugeruje obecność sezonowości w danych. Wzorce te odpowiadają pewnym cyklom czasowym, co może wskazywać na sezonową zmienność w ruchu turystycznym. Na wykresie wyraźnie widoczny jest trend wzrostowy, szczególnie do roku 2013. Obserwacja,

że zmienność rośnie wraz z poziomem szeregu, sugeruje możliwość występowania sezonowości multiplikatywnej. To znaczy, że zmienność nie jest stała w czasie, a jej wzrost zależy od poziomu szeregu czasowego.

### 1.3.2 Wykres sezonowy

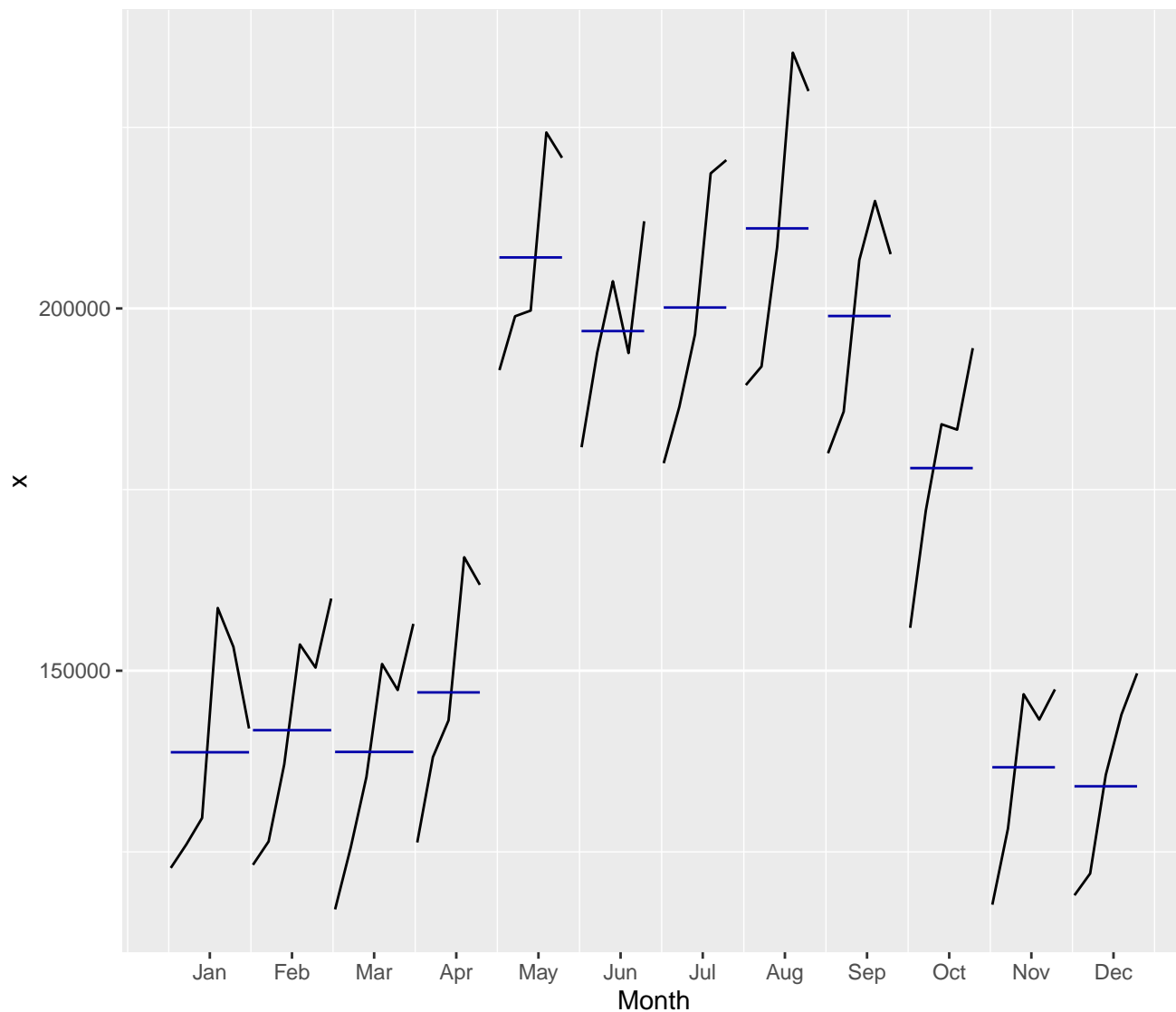


Rysunek 2: Wykres sezonowy dla danych szeregu czasowego - Hotele

#### Wnioski:

Wykres sugeruje silną sezonowość związaną z miesiącami. W okresie wiosennym obserwujemy wyraźny wzrost liczby gości hotelowych, osiągając maksimum w maju. Od maja do czerwca/lipca, notujemy niewielki spadek liczby gości. Później zauważamy ponowny wzrost aż do sierpnia, który charakteryzuje się najwyższą liczbą gości hotelowych w porównaniu do innych miesięcy, to może być związane z letnimi wakacjami i wzmożoną aktywnością turystyczną. Okres od września do listopada charakteryzuje się znacznym spadkiem. Po spadku w listopadzie, obserwujemy mały wzrost liczby gości w grudniu i styczniu. Wnioski te sugerują, że sezonowość w danych jest ściśle związana z porą roku.

### 1.3.3 Wykres miesięczny

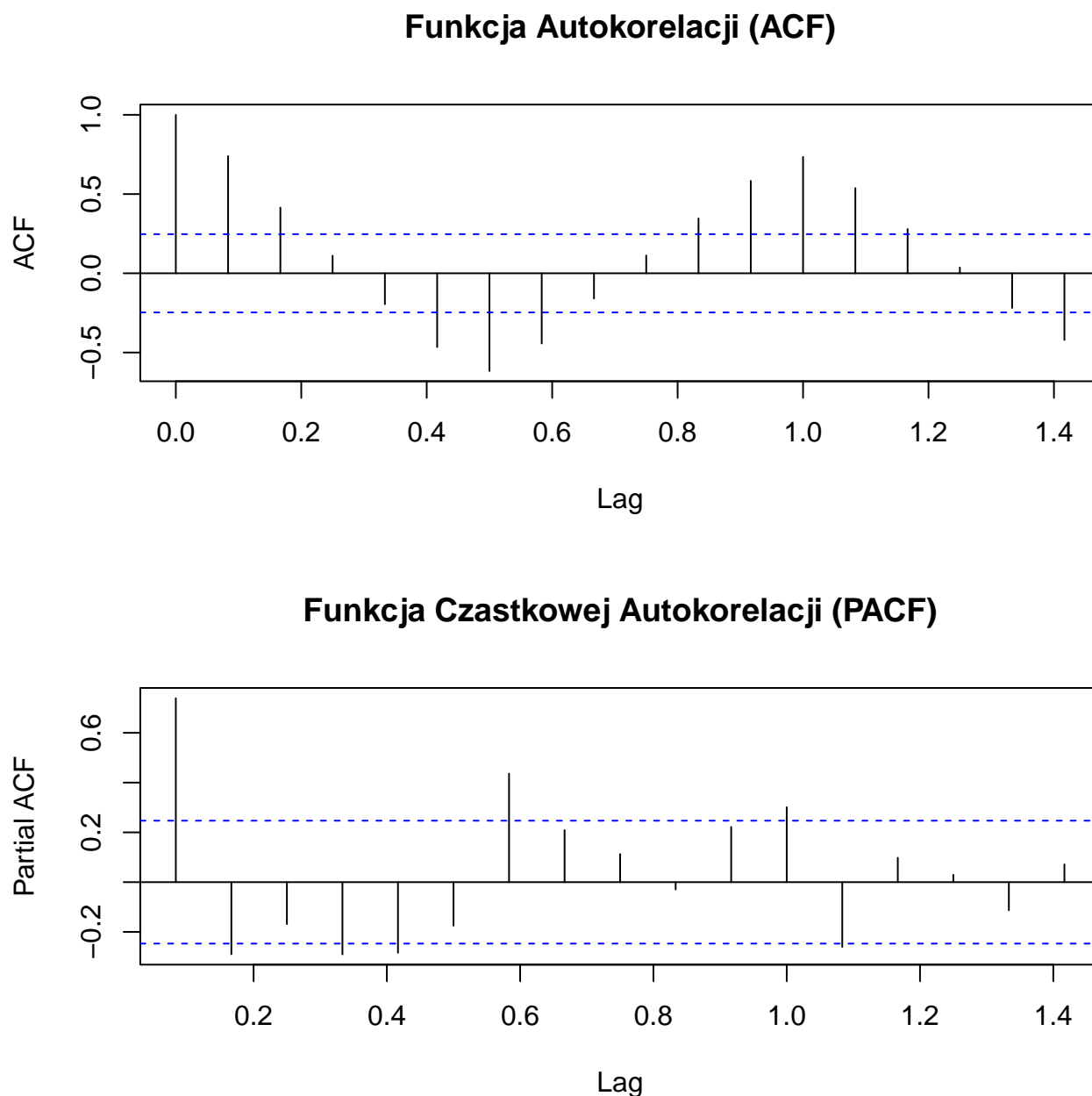


Rysunek 3: Wykres miesięczny dla danych szeregu czasowego - Hotele

Wnioski:

Wykres potwierdza wcześniejsze wnioski - wyraźne sezonowe zmiany w ruchu turystycznym. Miesiące cieplejsze, zwłaszcza lato, wykazują tendencję wzrostową, osiągając najwyższe wartości w sierpniu.

### 1.3.4 Wykres ACF i PACF



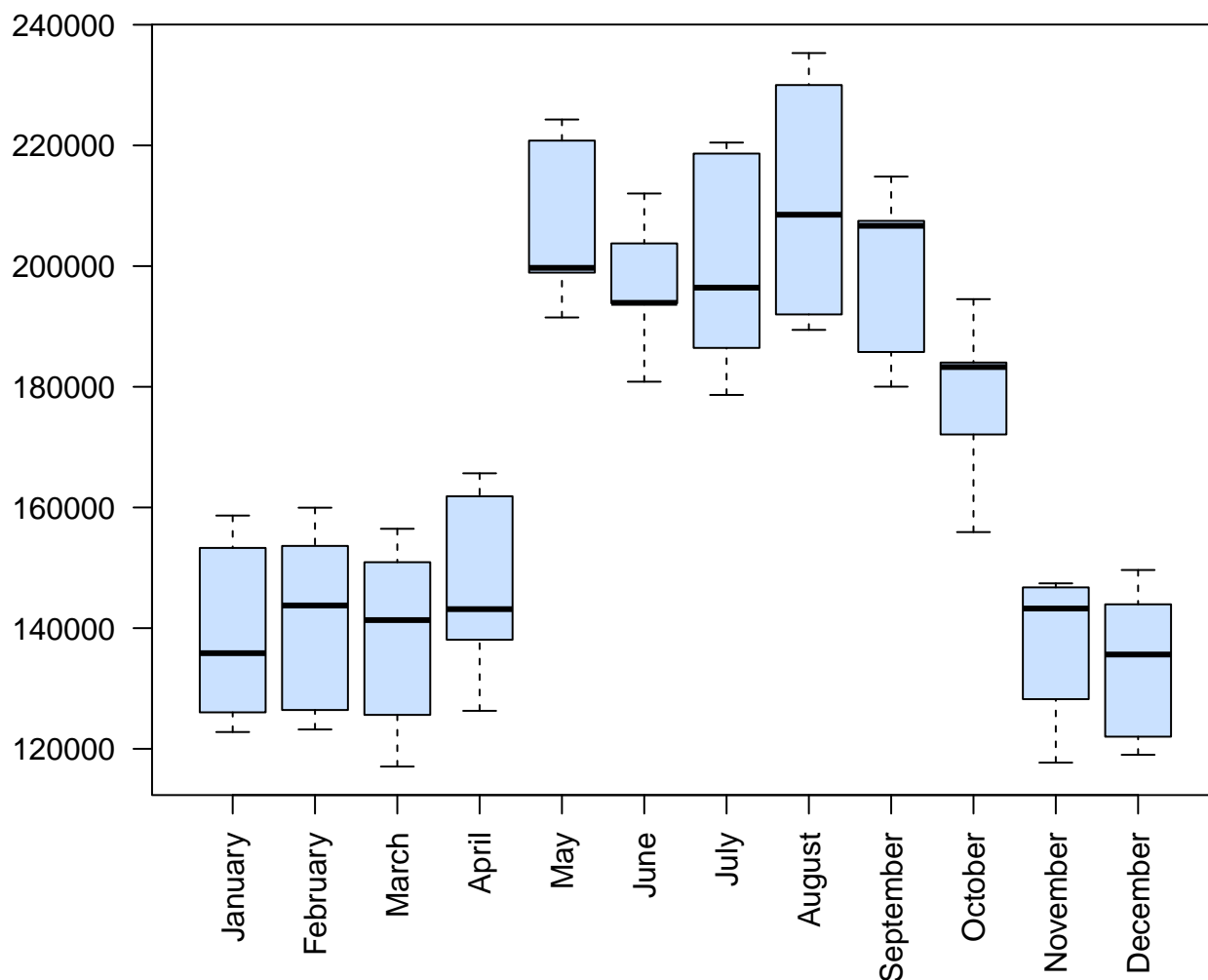
Rysunek 4: Wykresy ACF i PACF dla danych szeregu czasowego - Hotele

#### Wnioski:

Na wykresie ACF możemy zauważyć obecność regularnych wzorców spadku i wzrostu, potwierdza to sezonowość w danych. Ten rodzaj wzoru może wskazywać na sezonową zależność między obserwacjami. Obserwacja wartości funkcji autokorelacji wychodzących poza przedział ufności sugeruje, że dane zawierają pewne struktury niestandardowe, więc możemy wnioskować, że szereg czasowy nie jest białym szumem. Obecność sezonowości wskazuje na brak stacjonarności w szeregu czasowym.

### 1.3.5 Wykres pudełkowy

**Boxplot dla danych szeregu czasowego – Hotele**

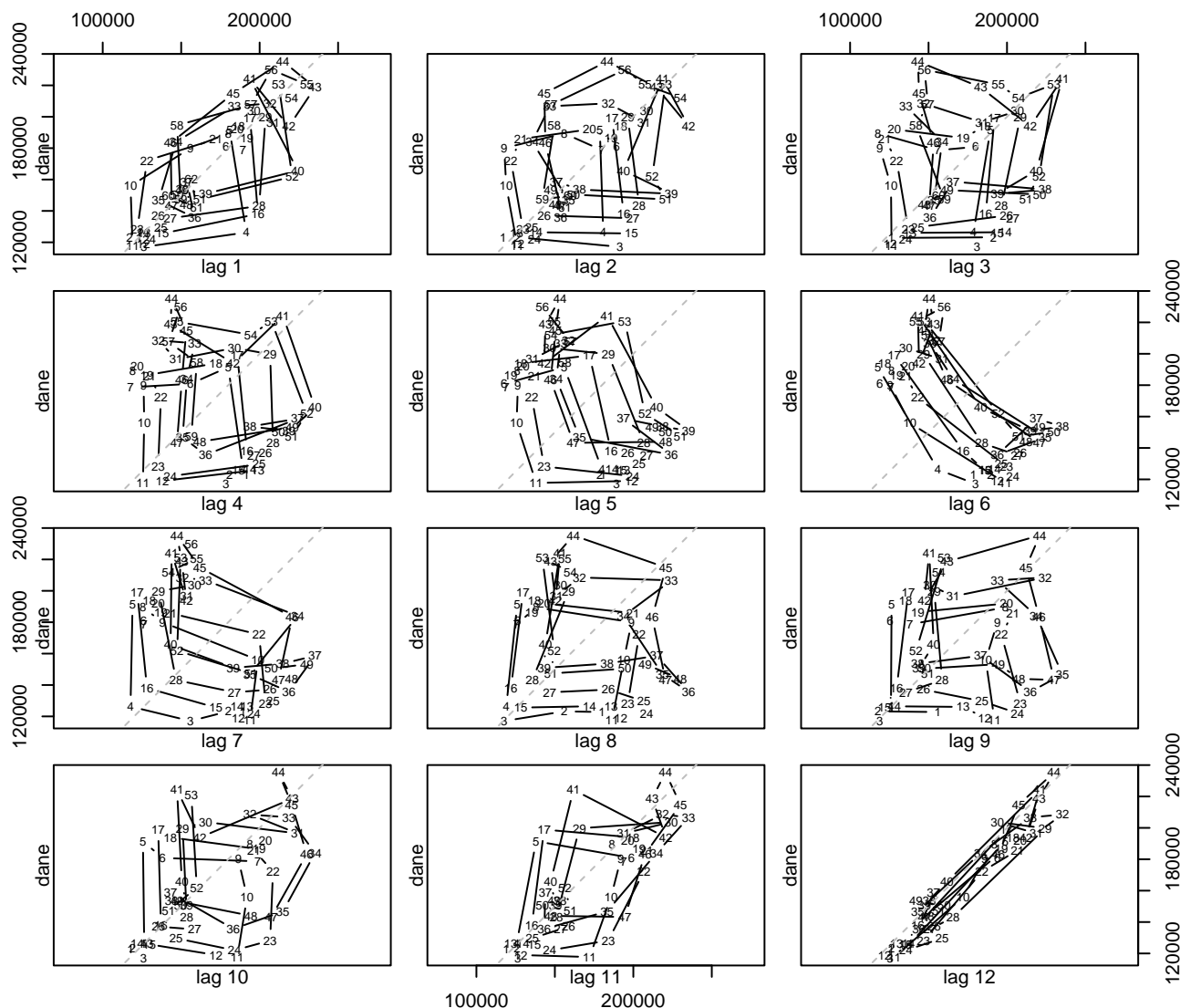


Rysunek 5: Wykres pudełkowy dla danych szeregu czasowego - Hotele

Wnioski:

Widoczne większe zakresy oraz wyższe wartości dla cieplejszych miesięcy potwierdzają sezonowość w danych, co jest zgodne z wcześniejszymi analizami. Brak wartości odstających wskazuje na to, że dane w poszczególnych miesiącach nie odbiegają znacząco od ogólnego wzorca sezonowego.

### 1.3.6 Wykres rozrzutu dla opóźnień $h=1,...,12$



Rysunek 6: Wykres rozrzutu dla danych szeregu czasowego - Hotele

Wnioski:

Wykres pokazuje, że największa korelacja występuje dla opóźnienia 12 miesięcy, co potwierdza sezonowość danych. Oznacza to, że obserwacje są silnie skorelowane z obserwacjami z tego samego miesiąca z poprzednich lat. Brak wyraźnych skorelowanych opóźnień poza 12 miesiącami może sugerować, że sezonowość stanowi dominujący czynnik wpływający na szereg czasowy.



## 1.4 Zastosowanie i porównanie metod dekompozycji

### 1.4.1 Dekompozycja na podstawie ruchomej średniej (funkcja `decompose()`)

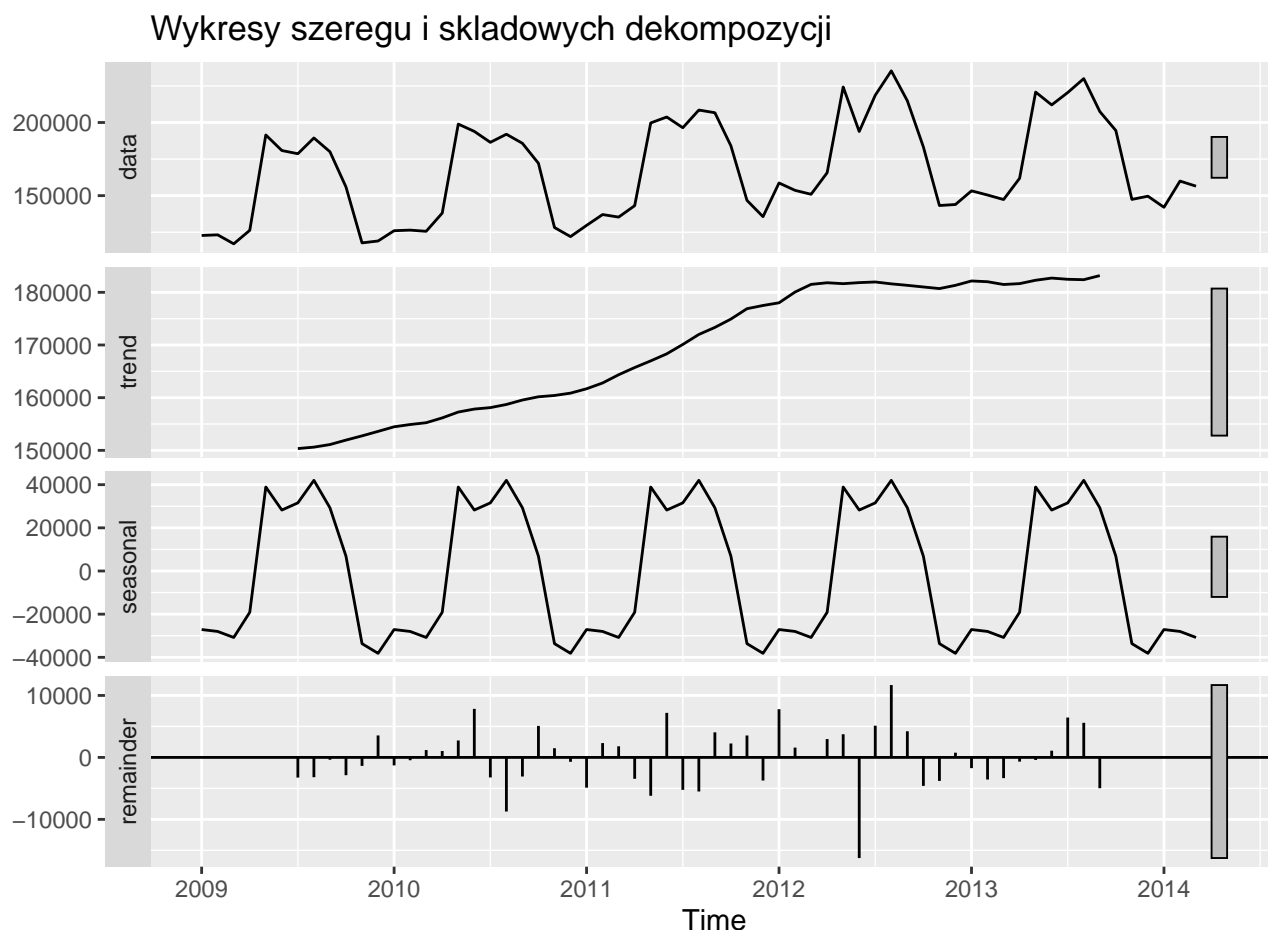
Przy pomocy funkcji `decompose()` przeprowadzimy analizę szeregu czasowego, wykonując dekompozycję zarówno wersji addytywnej, jak i multiplikatywnej przy użyciu ruchomej średniej. W obu przypadkach zidentyfikujemy składowe takie jak trend, sezonowość i reszty. Przedstawimy wykresy składowych oraz przeanalizujemy reszty pod kątem sezonowości, białego szumu i stacjonarności.

- Dekompozycja addytywna

```
# dekompozycja addytywna
dekomp.add <- decompose(dane, type="additive")

# dekompozycja addytywna - składowe
dekomp.add.trend <- dekomp.add$trend
dekomp.add.sezonowosc <- dekomp.add$seasonal
dekomp.add.reszty <- dekomp.add$random
```

1. Wykresy szeregu i składowych dekompozycji



Rysunek 7: Wykresy szeregu i składowych - dekompozycja addytywna

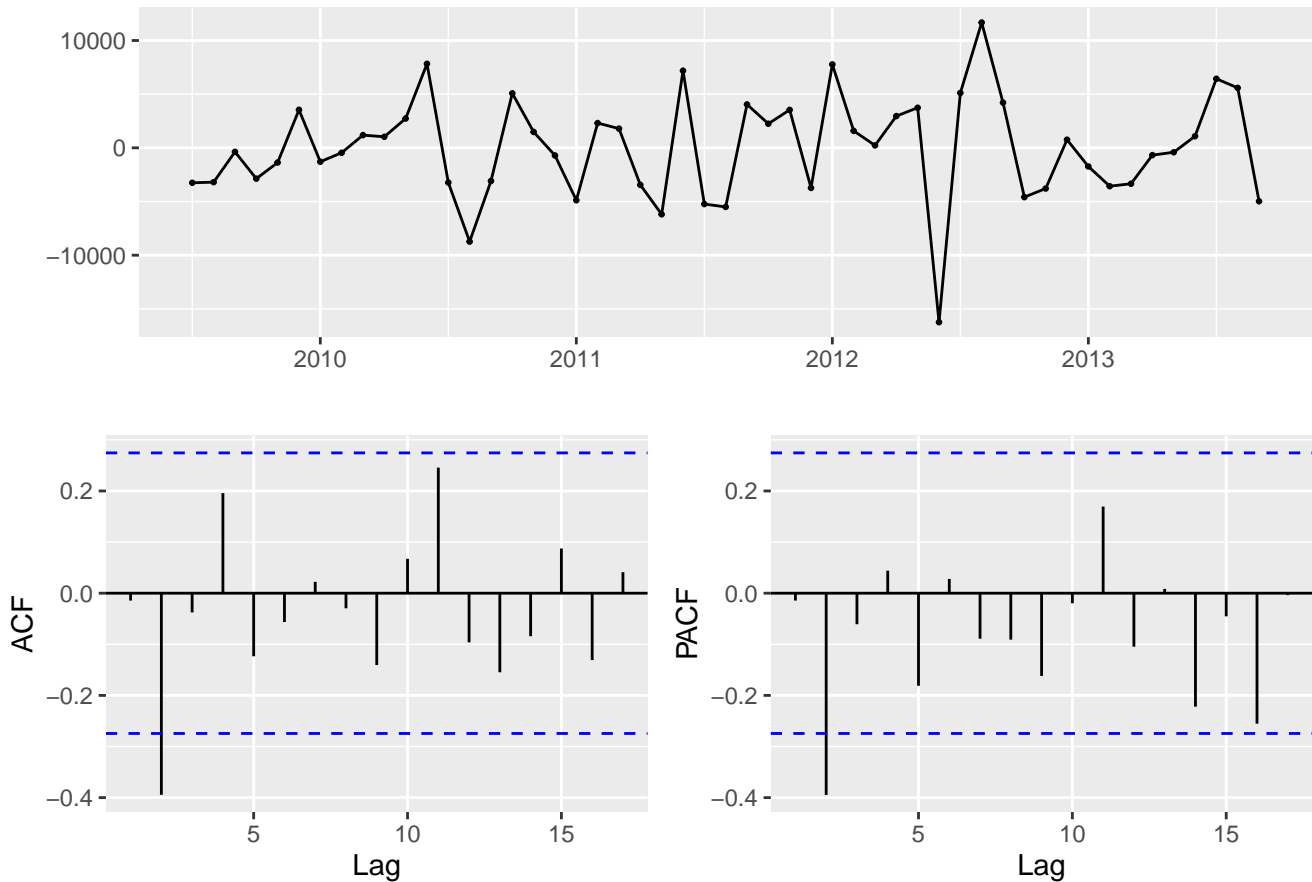
Wnioski:

Analiza wykresu dekompozycji addytywnej potwierdza, że szereg czasowy charakteryzuje się:

Rosnącym Trendem - wartości trendu rosną w miarę upływu czasu.  
 Obecnością Sezonowości - występują regularne wzorce sezonowe w danych.  
 Te obserwacje potwierdzają wcześniejsze wnioski dotyczące charakterystyki szeregu czasowego.

## 2. Wykres reszt

Następnie zajmujemy się analizą reszt uzyskanych w wyniku dekompozycji addytywnej szeregu czasowego. Reszty w kontekście analizy szeregów czasowych to pozostałości po odjęciu składowych trendu i sezonowości od oryginalnego szeregu danych. Poniżej wyświetlamy wykres reszt.



Rysunek 8: Wykres reszt - dekompozycja addytywna

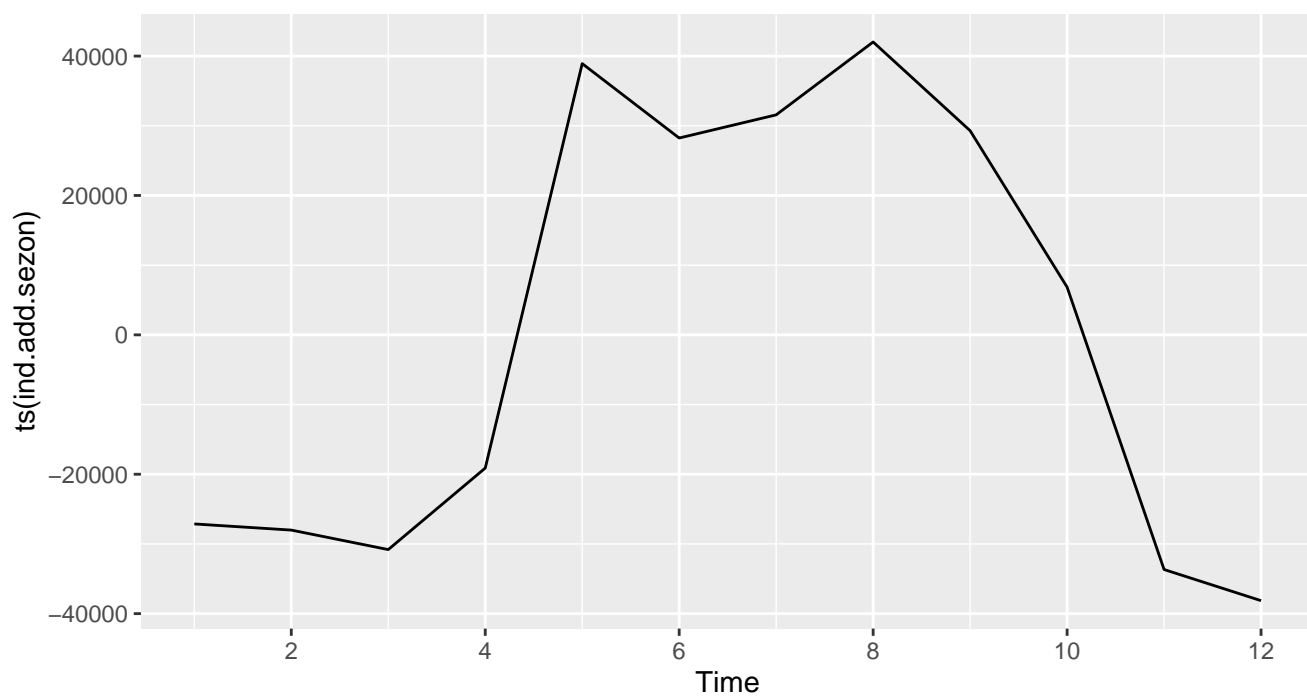
### Wnioski:

Wykres reszt ukazuje, że istnieją pewne pozostałości sezonowości, co sugeruje, że dekompozycja addytywna nie całkowicie eliminuje sezonowości z danych.

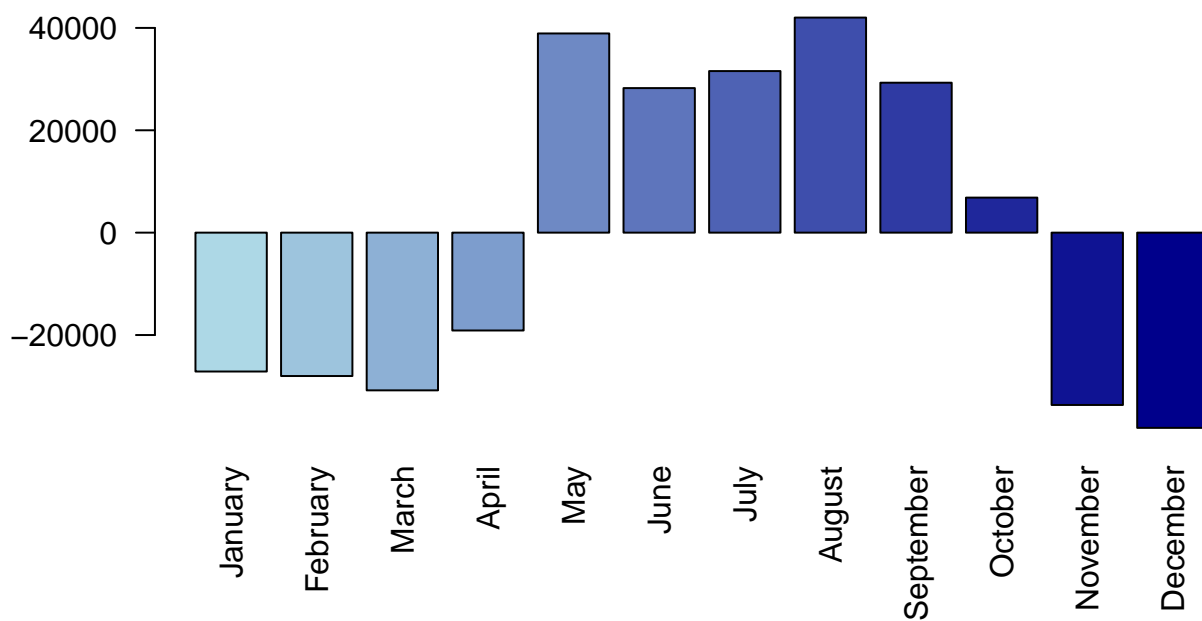
Reszty nie są białym szumem, ponieważ na wykresie ACF jest za dużo wartości poza przedziałem ufności. Analiza wariancji reszt wskazuje, że nie ma jednorodności wariancji. Wariancja reszt zmienia się w miarę upływu czasu, z mniejszymi wartościami w pewnych okresach, co sugeruje brak stacjonarności w danych.

## 3. Indykator sezonowości

W kolejnym etapie analizy skupimy się na badaniu indykatora sezonowości, uzyskanego w wyniku dekompozycji addytywnej szeregu czasowego. Indykator ten stanowi istotny element, pozwalający zidentyfikować wzorce sezonowe w danych. Poniżej przedstawiamy wizualizację indykatora sezonowości za pomocą wykresu czasowego oraz wykresu słupkowego dla poszczególnych miesięcy.



Rysunek 9: Wykresy indykatora sezonowości - dekompozycja addytywna



Rysunek 10: Wykresy indykatora sezonowości - dekompozycja addytywna

### Wnioski:

Na wykresie czasowym indikatora sezonowości można zauważyć wyraźne wzrosty w określonych miesiącach. Analiza wykresu słupkowego dla poszczególnych miesięcy ujawnia ewidentne wzorce sezonowe. Obserwujemy regularne spadki zimą i wzrosty latem, co potwierdza obecność sezonowości w danych.

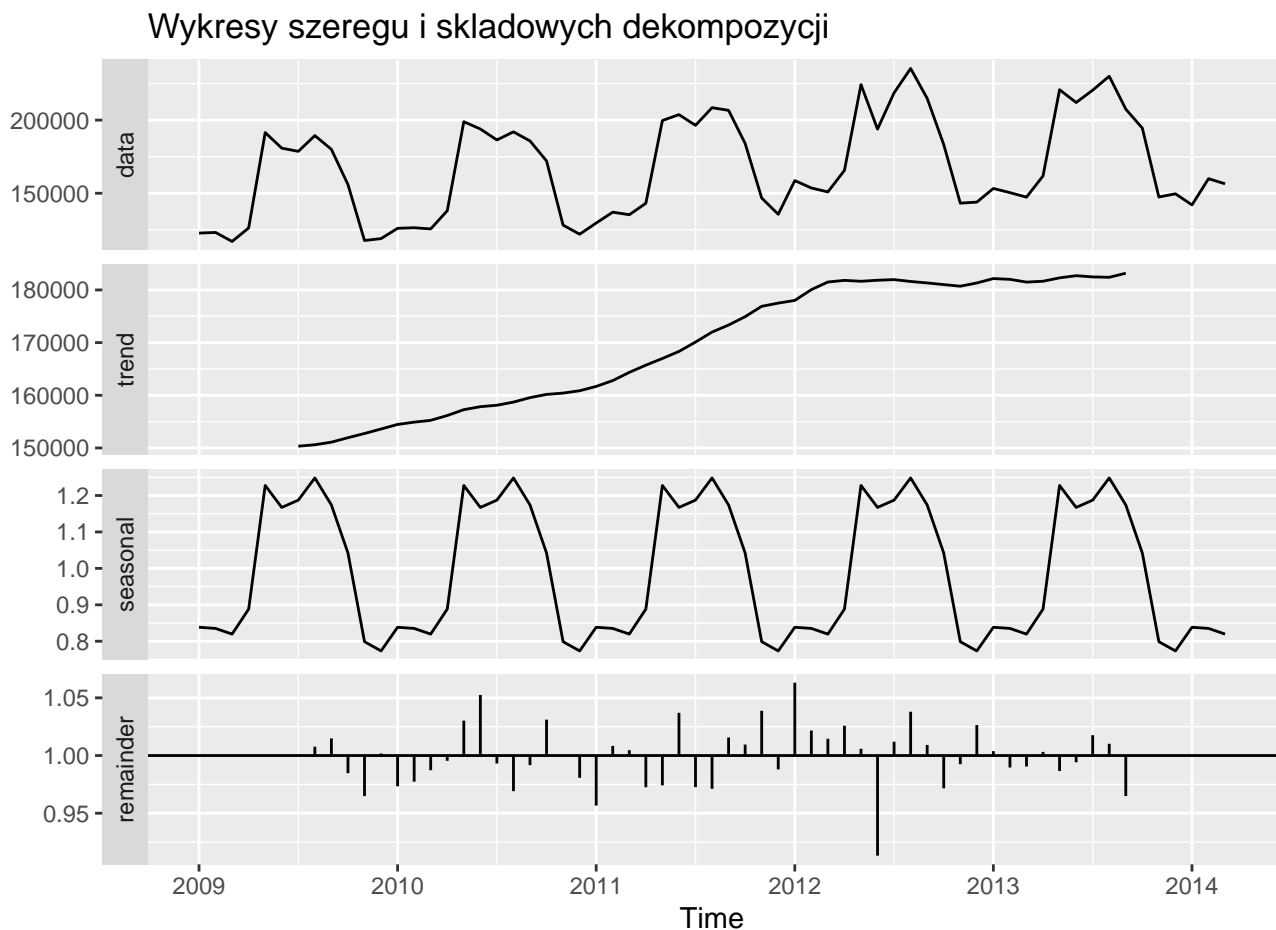
- Dekompozycja multiplikatywna

Dekompozycja multiplikatywna polega na tym, że rozkładamy szereg w taki sposób, że jest równy iloczynowi trendu, sezonowości i losowej reszty. Reszty w dekompozycji multiplikatywnej są zwykle bliskie 1, w przeciwieństwie do dekompozycji addytywnej, gdzie były one bliskie zera. W związku z tym, poziomem odniesienia dla składowej sezonowej w dekompozycji multiplikatywnej jest jedynka.

```
# dekompozycja multiplikatywna
dekomp.mul <- decompose(x=dane, type="multiplicative")

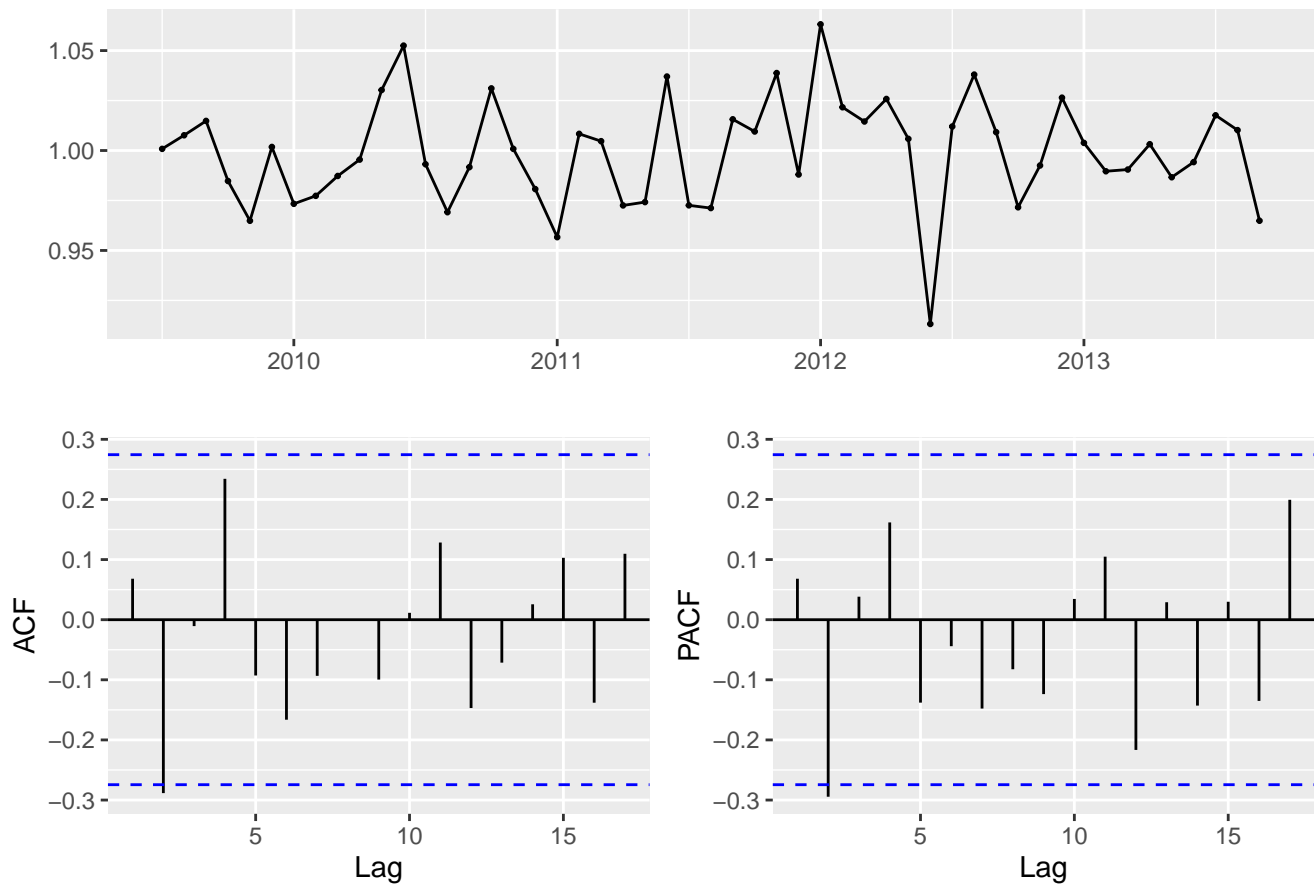
# dekompozycja multiplikatywna - składowe
dekomp.mul.trend <- dekomp.mul$trend
dekomp.mul.sezonowosc <- dekomp.mul$seasonal
dekomp.mul.reszty <- dekomp.mul$random
```

1. Wykresy szeregu i składowych dekompozycji



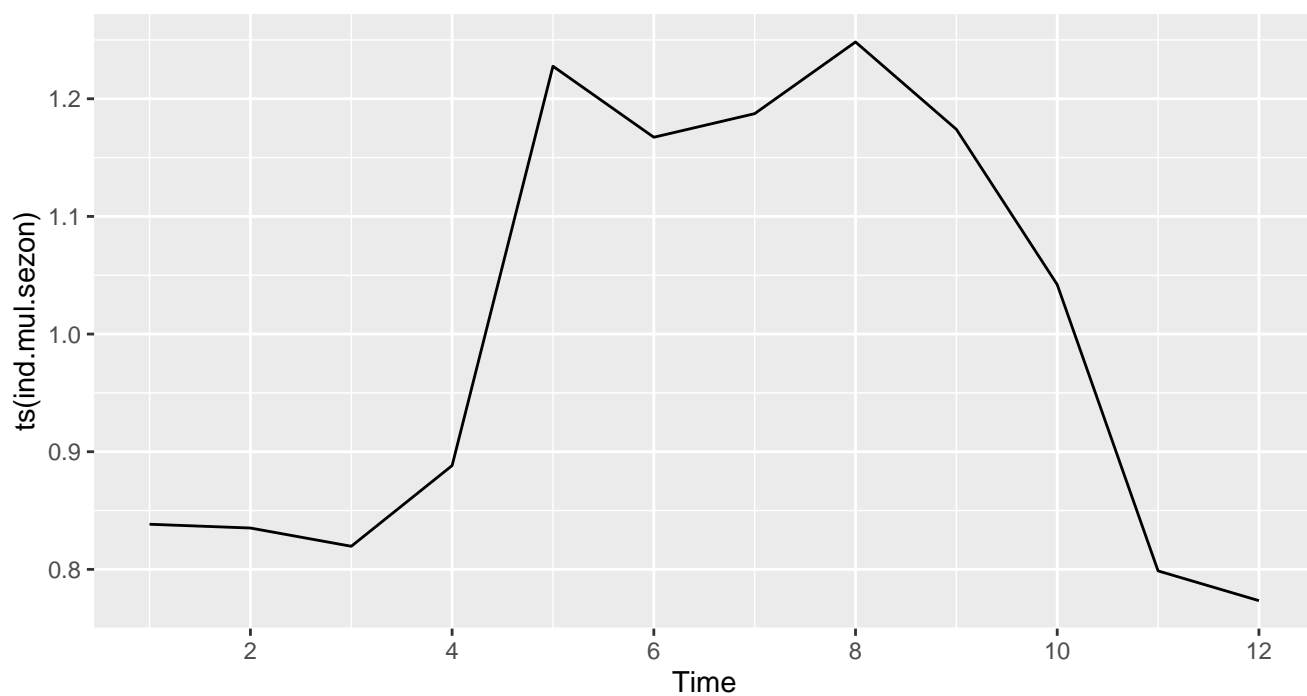
Rysunek 11: Wykresy szeregu i składowych - dekompozycja multiplikatywna

## 2. Wykres reszt

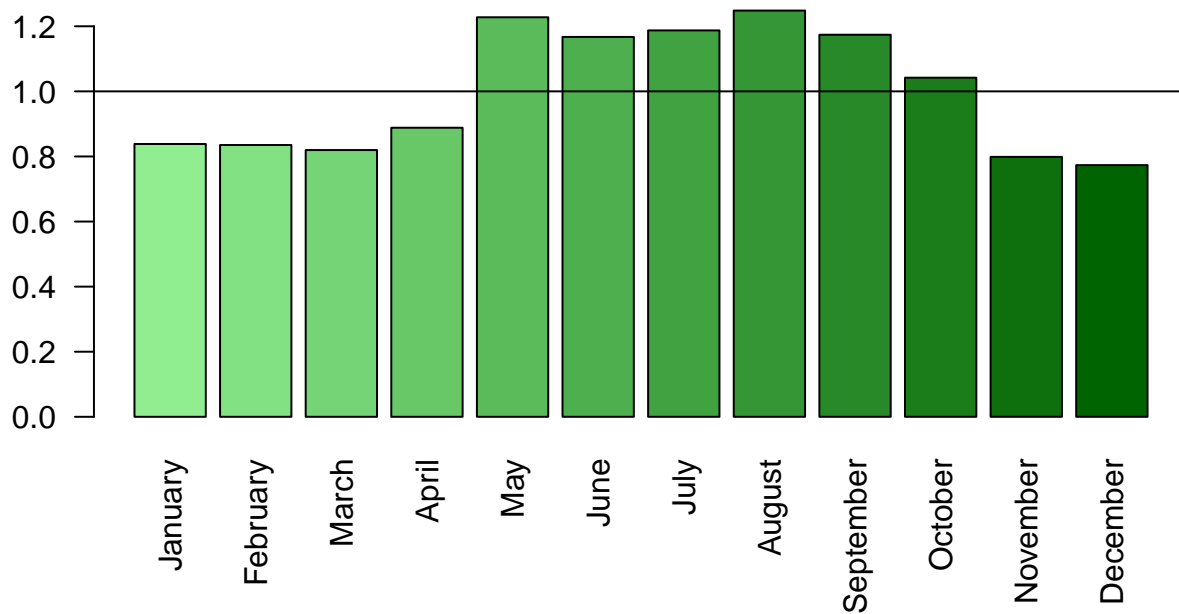


Rysunek 12: Wykres reszt - dekompozycja multiplikatywna

## 3. Indykator sezonowości



Rysunek 13: Wykresy indykatora sezonowości - dekompozycja multiplikatywna



Rysunek 14: Wykresy indykatora sezonowości - dekompozycja multiplikatywna

Wnioski:

Analiza dekompozycji multiplikatywnej wskazuje, że struktura szeregu czasowego jest podobna do dekompozycji addytywnej. Oba podejścia dostarczają zbliżone wyniki, co może sugerować, że wybór między nimi nie ma kluczowego znaczenia w analizie tego konkretnego szeregu czasowego.

Zobaczmy, jak dane dekompozycje addytywna i multiplikatywna poradziły sobie z dopasowaniem do naszych danych.



Rysunek 15: Wykres dopasowania modeli dekompozycji na podstawie ruchomej średniej

Model addytywny zakłada, że składniki składowe (trend, sezonowość) są dodawane do siebie. Wydaje się, że dekompozycja addytywna nie radzi sobie idealnie z dopasowaniem do danych. W niektórych miejscach występują różnice między modelem a rzeczywistymi danymi, co sugeruje, że ten model może niedoszacowywać pewnych nieregularności.

Model multiplikatywny zakłada, że składniki składowe są mnożone przez siebie nawzajem. W porównaniu z dekompozycją addytywną, dekompozycja multiplikatywna zdaje się lepiej dostosowywać do danych. Krzywe dopasowania są bliższe rzeczywistym danym. Model ten może lepiej uwzględniać zmienność wzorców sezonowych i bardziej stabilnie oddać nieregularności.

W związku z powyższym, dekompozycja multiplikatywna wydaje się być bardziej odpowiednia dla analizowanego szeregu czasowego, gdyż lepiej odwzorowuje zmienność sezonową i jest bardziej precyzyjna w modelowaniu różnic między modelem a rzeczywistymi danymi. Zatem dekompozycja multiplikatywna będzie reprezentowała dekompozycję na podstawie ruchomej średniej w porównaniu metod dekompozycji szeregu czasowego.

#### 1.4.2 Dekompozycja na podstawie modelu regresji (funkcja `tslm()`)

Przy pomocy funkcji `tslm()` przeprowadzimy analizę szeregu czasowego, wykonując dekompozycję na podstawie modelu regresji dla różnych modeli parametrycznych dla trendu (np. wielomiany różnych stopni).

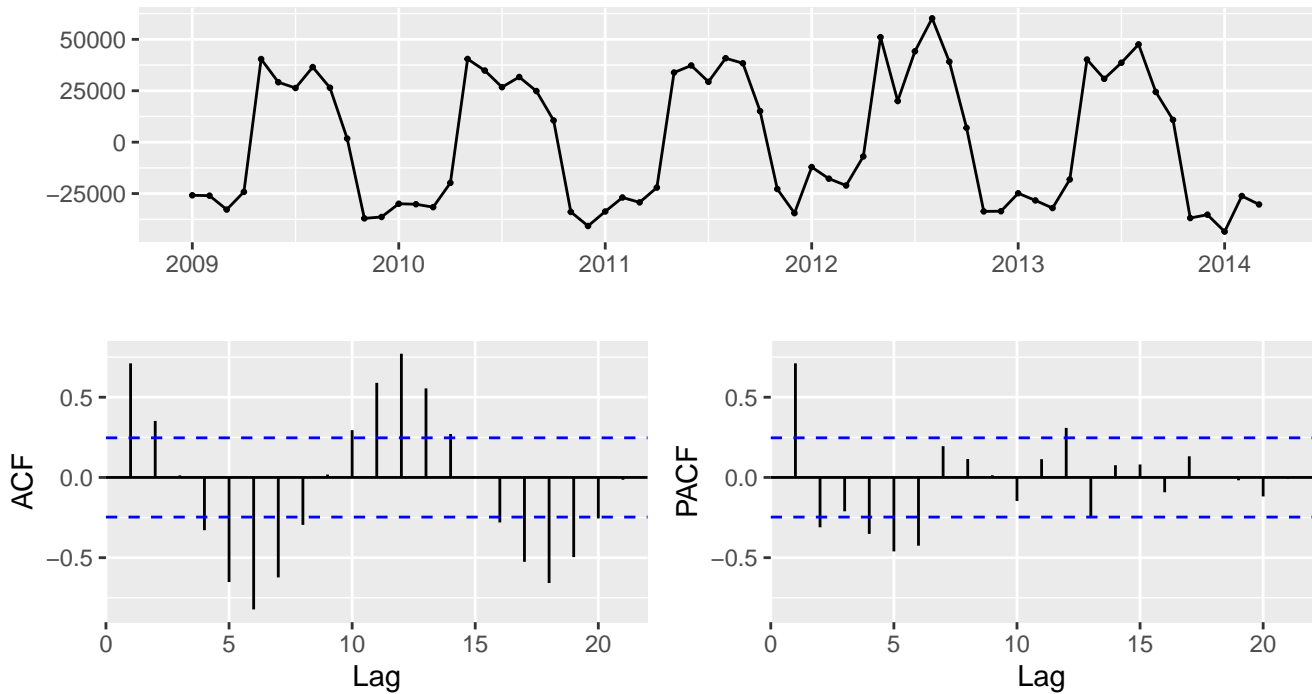
- Trend liniowy

```
model1 <- tslm(dane~trend)
```

Wyznamy szczególowe informacje dotyczace modelu, takie jak wspólcynniki regresji, wartosci t-statystyki,  $R^2$  za pomoca funkcji *summary*.

```
[1] 0.1109241
```

Sporzadzimy wykres diagnostyczny dla danego modelu szeregu czasowego.



Rysunek 16: Wykres diagnostyczny: model 1, trend liniowy

Wnioski:

- Nasz model to:  $\text{dane} = 148006.1 + 615.2 * \text{trend}$
- Wartość  $R^2$  dla trendu liniowego wynosi 0.1109. Oznacza to, że około 11.09% zmienności w danych może być wyjaśnione przez model liniowy. Wartość ta nie jest bardzo wysoka, co sugeruje, że trend liniowy nie tłumaczy znacznej części zmienności w danych.
- Test istotności dla obu współczynników modelu pokazuje, czy te współczynniki są statystycznie istotne.
- Na wykresie diagnostycznym widac pozostałości sezonowości, trend w zasadzie jest usunięty.

- Trend liniowy + sezonowość

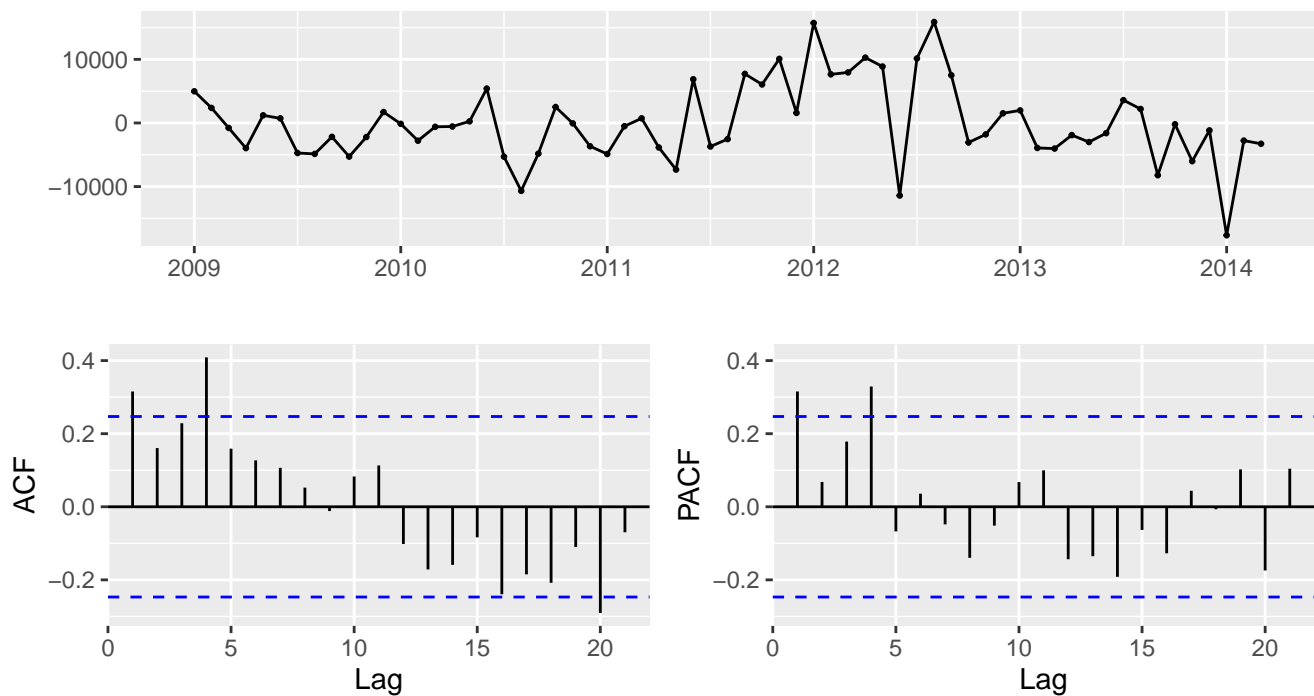
```
model2 <- tslm(dane~trend+season)
```

Wyznamy szczególowe informacje dotyczace modelu, takie jak wspólcynniki regresji, wartosci t-statystyki,  $R^2$  za pomoca funkcji *summary*.

```
[1] 0.967188
```



Sporządźmy wykres diagnostyczny dla danego modelu szeregu czasowego.



Rysunek 17: Wykres diagnostyczny: model 1, trend liniowy

Wnioski:

- Nasz model to:  $\text{dane} = 117103.44 + 698.14 * \text{trend} + \sum_{i=2}^{12} \text{season}_i$
- Wartość  $R^2$  dla modelu z trendem liniowym i sezonowością wynosi 0.9672. Oznacza to, że około 96.72% zmienności w danych może być wyjaśnione przez ten model. To znacznie lepsze dopasowanie niż w przypadku modelu z samym trendem liniowym.
- Wszystkie współczynniki modelu są uznawane za istotne statystycznie, z wyjątkiem niektórych sezonów (season2, season3, season11, season12), które nie są istotne. Oznacza to, że te sezonowe składniki nie mają statystycznie istotnego wpływu na szereg czasowy.

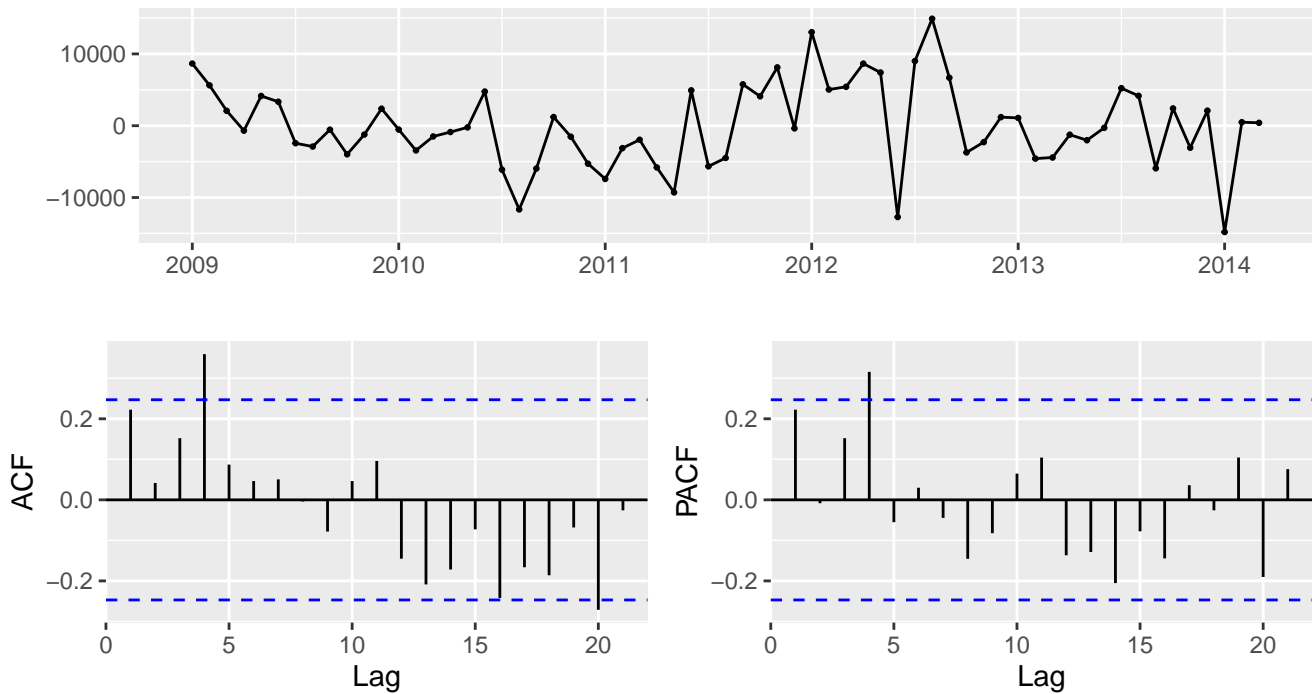
- Trend kwadratowy + sezonowość

```
model3 <- tslm(dane~trend+season+I(trend^2))
```

Wyznamy szczegółowe informacje dotyczące modelu, takie jak współczynniki regresji, wartości t-statystyki,  $R^2$  za pomocą funkcji *summary*.

```
[1] 0.9706345
```

Sporządźmy wykres diagnostyczny dla danego modelu szeregu czasowego.



Rysunek 18: Wykres diagnostyczny: model 1, trend liniowy

Wnioski:

- Nasz model to:  $\text{dane} = 113004.087 + 1133.225 * \text{trend} - 6.798 * \text{trend}^2 + \sum_{i=2}^{12} \text{season}_i$
- Wartość  $R^2$  dla modelu z trendem kwadratowym i sezonowością wynosi 0.9706. Oznacza to, że około 97.06% zmienności w danych może być wyjaśnione przez ten model. To jeszcze lepsze dopasowanie niż w przypadku modelu z trendem liniowym i sezonowością.
- Wszystkie współczynniki modelu są uznawane za istotne statystycznie, z wyjątkiem niektórych sezonów (season2, season3, season11), które nie są istotne. Oznacza to, że te sezonowe składniki nie mają statystycznie istotnego wpływu na szereg czasowy. Współczynnik przy  $\text{trend}^2$  jest również istotny statystycznie, co sugeruje, że wprowadzenie składnika kwadratowego do modelu jest uzasadnione.

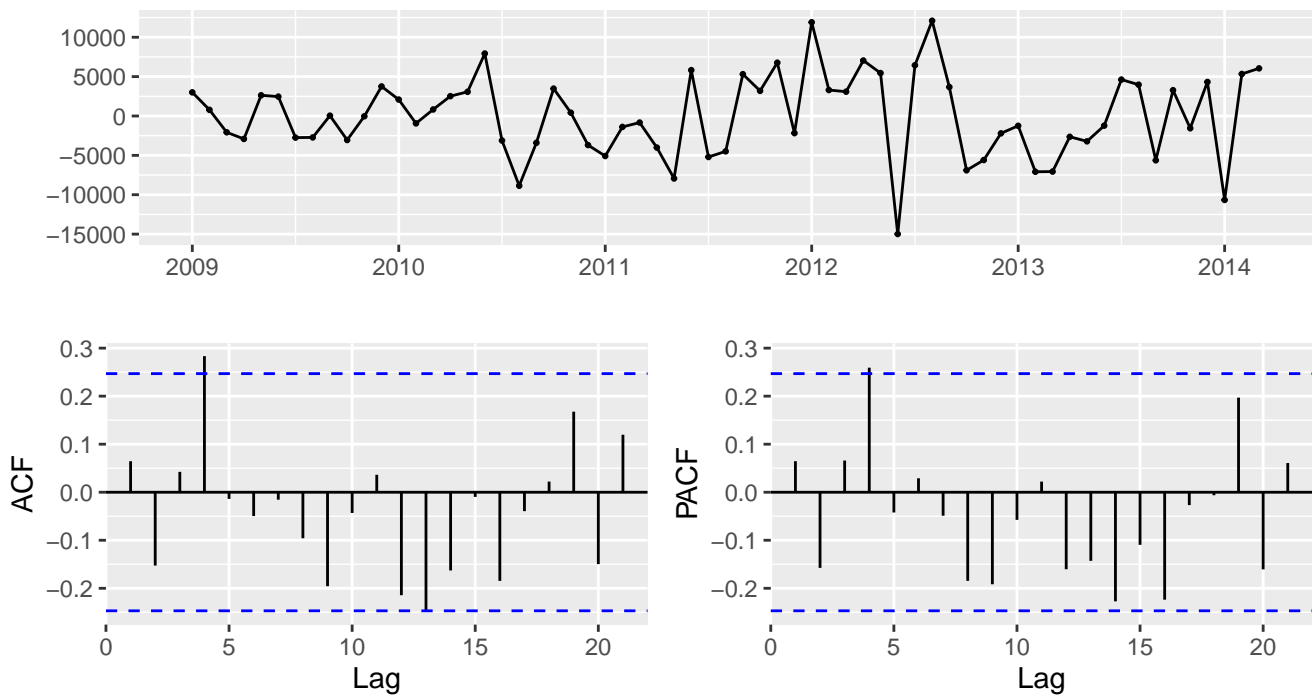
- Trend sześcienny + sezonowość

```
model4 <- tslm(dane~season + poly(trend, raw = FALSE, degree = 3))
```

Wyznamy szczegółowe informacje dotyczące modelu, takie jak współczynniki regresji, wartości t-statystyki,  $R^2$  za pomocą funkcji `summary`.

[1] 0.9758995

Sporządzimy wykres diagnostyczny dla danego modelu szeregu czasowego.



Rysunek 19: Wykres diagnostyczny: model 1, trend liniowy

Wnioski:

- Nasz model to:  $\text{dane} = 139707.8 + 100425.3 * \text{trend} - 15952.9 * \text{trend}^2 - 19551.8 * \text{trend}^3 + \sum_{i=2}^{12} \text{season}_i$
- Wartość  $R^2$  dla modelu z trendem sześciennym i sezonowością wynosi 0.9759. Oznacza to, że około 97.59% zmienności w danych może być wyjaśnione przez ten model. Jest to bardzo wysoka wartość, co wskazuje na doskonałe dopasowanie modelu do danych.
- Wszystkie współczynniki modelu są uznawane za istotne statystycznie, z wyjątkiem niektórych sezonów (season2, season3, season11), które nie są istotne. Zarówno sezonowe składniki, jak i składniki trendu sześciennego mają istotny wpływ na szereg czasowy.

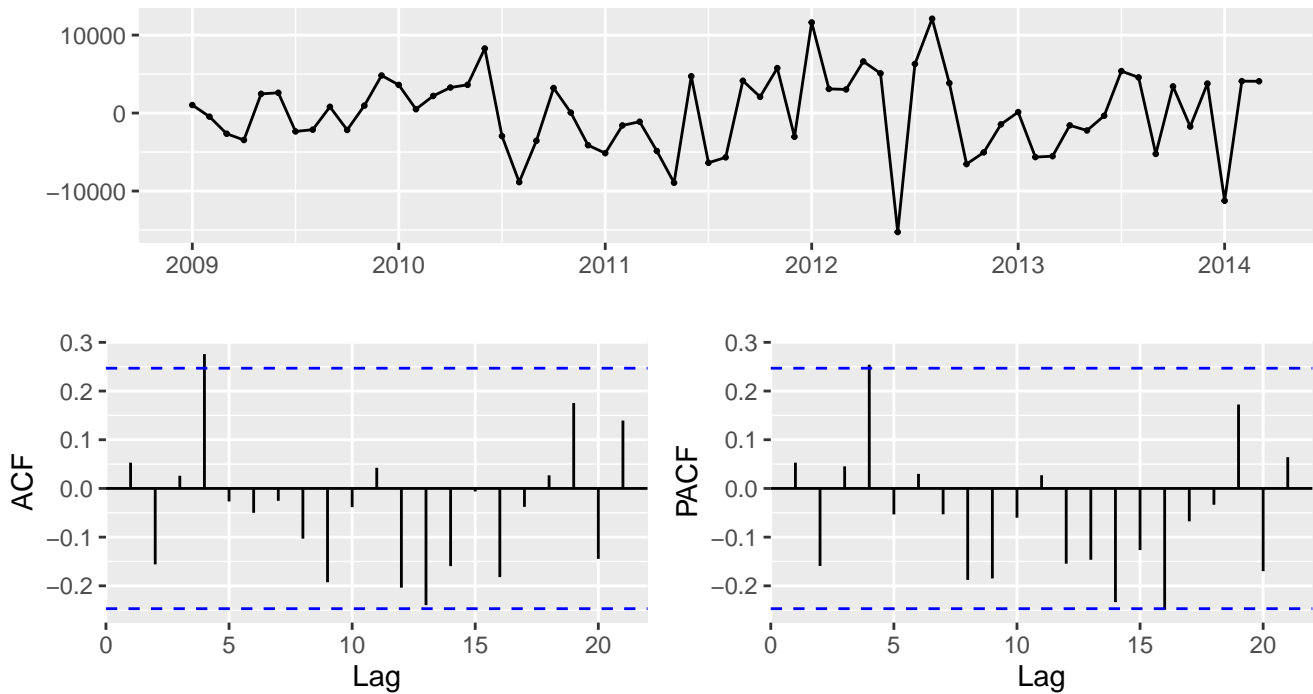
- Trend stopnia 4 + sezonowość

```
model5 <- tslm(dane~season + poly(trend, raw = FALSE, degree = 4))
```

Wyznamy szczegółowe informacje dotyczące modelu, takie jak współczynniki regresji, wartości t-statystyki,  $R^2$  za pomocą funkcji `summary`.

[1] 0.9765452

Sporządzimy wykres diagnostyczny dla danego modelu szeregu czasowego.



Rysunek 20: Wykres diagnostyczny: model 1, trend liniowy

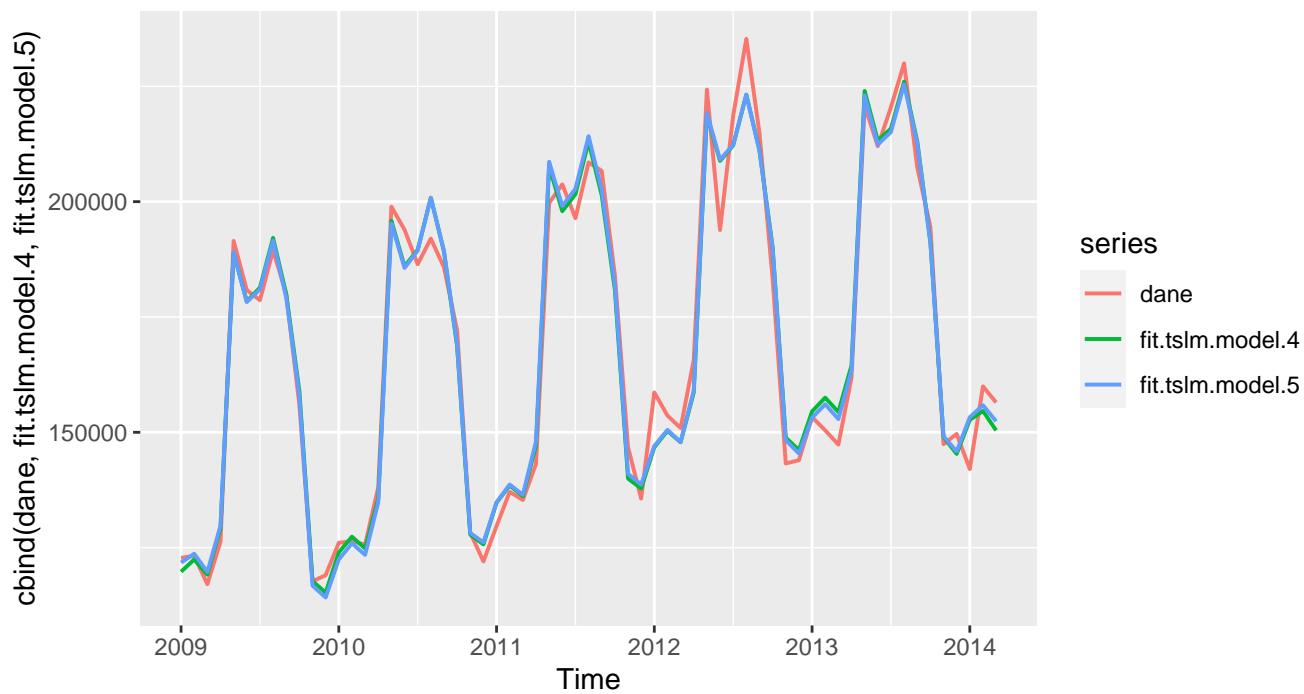
Wnioski:

- Nasz model to:  $\text{dane} = 139279.5 + 100425.3 * \text{trend} - 15538.3 * \text{trend}^2 - 19551.8 * \text{trend}^3 + 7098.3 * \text{trend}^4 + \sum_{i=2}^{12} \text{season}_i$
- Wartość  $R^2$  dla modelu z trendem stopnia 4 i sezonowością wynosi 0.9765. Oznacza to, że około 97.65% zmienności w danych może być wyjaśnione przez ten model. Jest to bardzo wysoka wartość, ale ona jest tylko nieco większa dla modelu z trendem sześciennym.
- Wszystkie współczynniki modelu są uznawane za istotne statystycznie, z wyjątkiem niektórych sezonów (season2, season3, season11), które nie są istotne. Zarówno sezonowe składniki, jak i składniki trendu stopnia 4 mają istotny wpływ na szereg czasowy.

Zobaczmy, jak te cztery modele poradziły sobie z dopasowaniem do naszych danych.



Rysunek 21: Wykres dopasowania modeli dekompozycji na podstawie modelu regresji



Rysunek 22: Wykres dopasowania modeli dekompozycji na podstawie modelu regresji

Można zauważyć, że modele 3 i 4 najbardziej się dopasowały. Porównujemy je osobno na wykresie niżej.



Rysunek 23: Wykres dopasowania modeli dekompozycji na podstawie modelu regresji

Model 4 (czyli trend sześcienny + sezonowość) najlepiej się dopasował do naszych danych i będzie reprezentował dekompozycję STL w porównaniu metod dekompozycji szeregu czasowego.

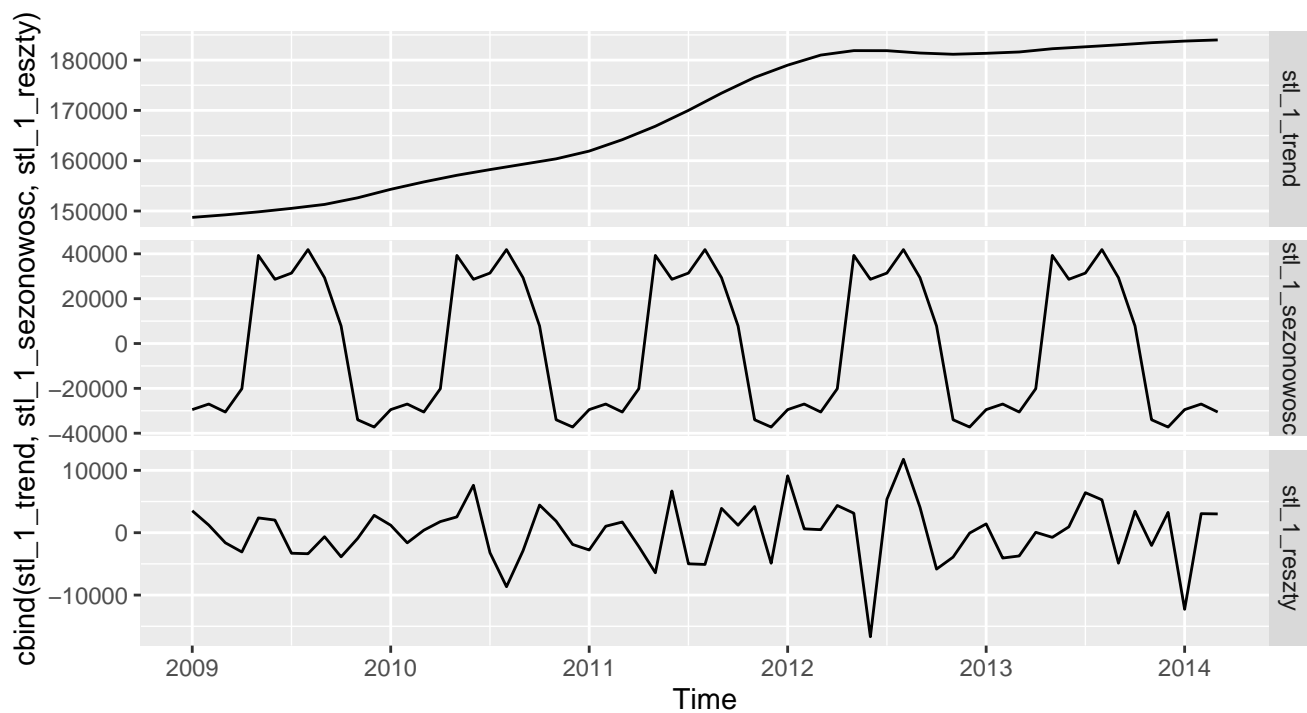
#### 1.4.3 Dekompozycja STL oparta na metodzie loess (funkcja `stl()`)

Przy pomocy funkcji `stl()` przeprowadzimy analizę szeregu czasowego, wykonując dekompozycję na podstawie metody loess dla różnych parametrów wygładzających *s.window* i *t.window*.

- Model 1: *s.window*='periodic', *t.window* - domyślne

```
stl1 <- stl(dane, s.window="periodic")  
  
stl_1_trend <- trendcycle(stl1)  
stl_1_sezonowosc <- seasonal(stl1)  
stl_1_reszty <- remainder(stl1)
```

Wykres składowych zamieszczamy na rysunku niżej.



Rysunek 24: Wykres składowych: stl, model 1

Dostajemy stały wzorec wahań sezonowych w kolejnych latach, czyli sezonowość jest funkcją okresową o okresie 12.

Sprawdzimy następnie, jak dany model dopasował się do naszych danych.



Rysunek 25: Wykres dopasowania modelu do danych: stl, model 1

Model dobrze oddaje ogólny trend i sezonowość, ale można zauważyć pewną niedoszacowaną zmienność w danych, co sugeruje, że pewne nieregularności nie zostały uwzględnione.

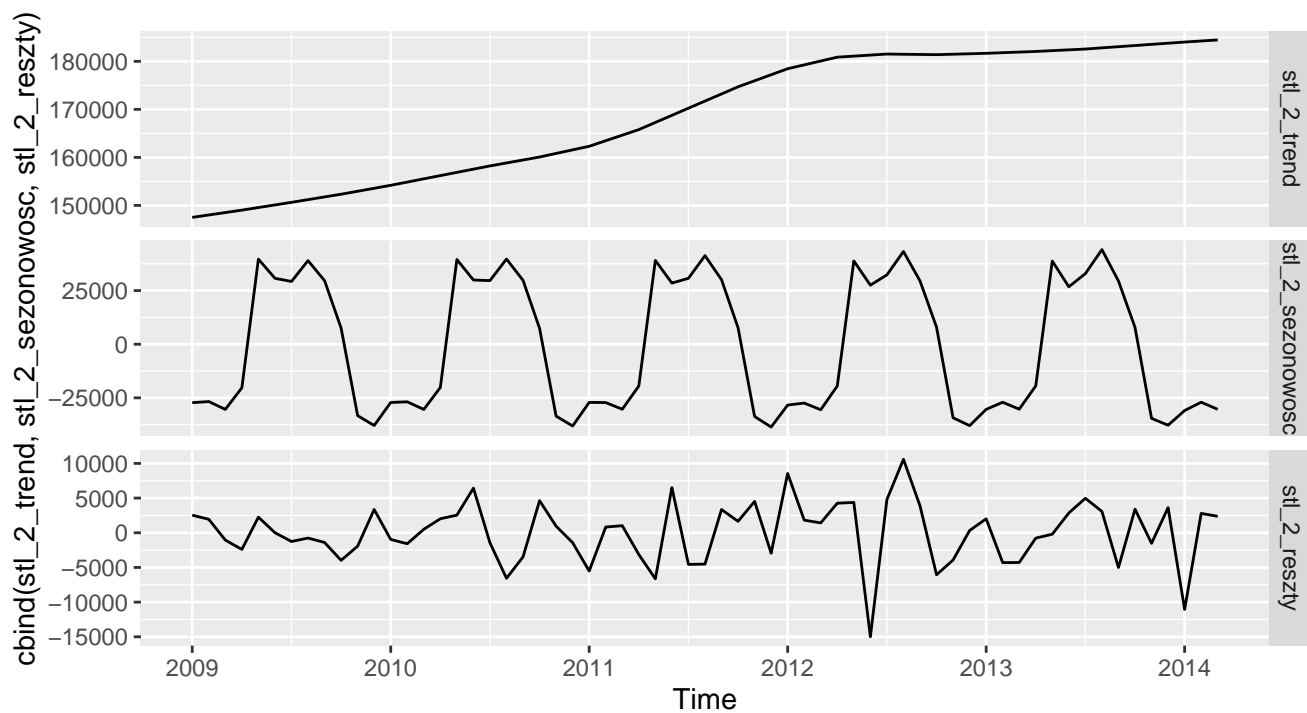
- Model 2: s.window=7, t.window - domyślne

```
stl2 <- stl(dane, s.window=7)

stl_2_trend <- trendcycle(stl2)
stl_2_sezonowosc <- seasonal(stl2)
stl_2_reszty <- remainder(stl2)
```

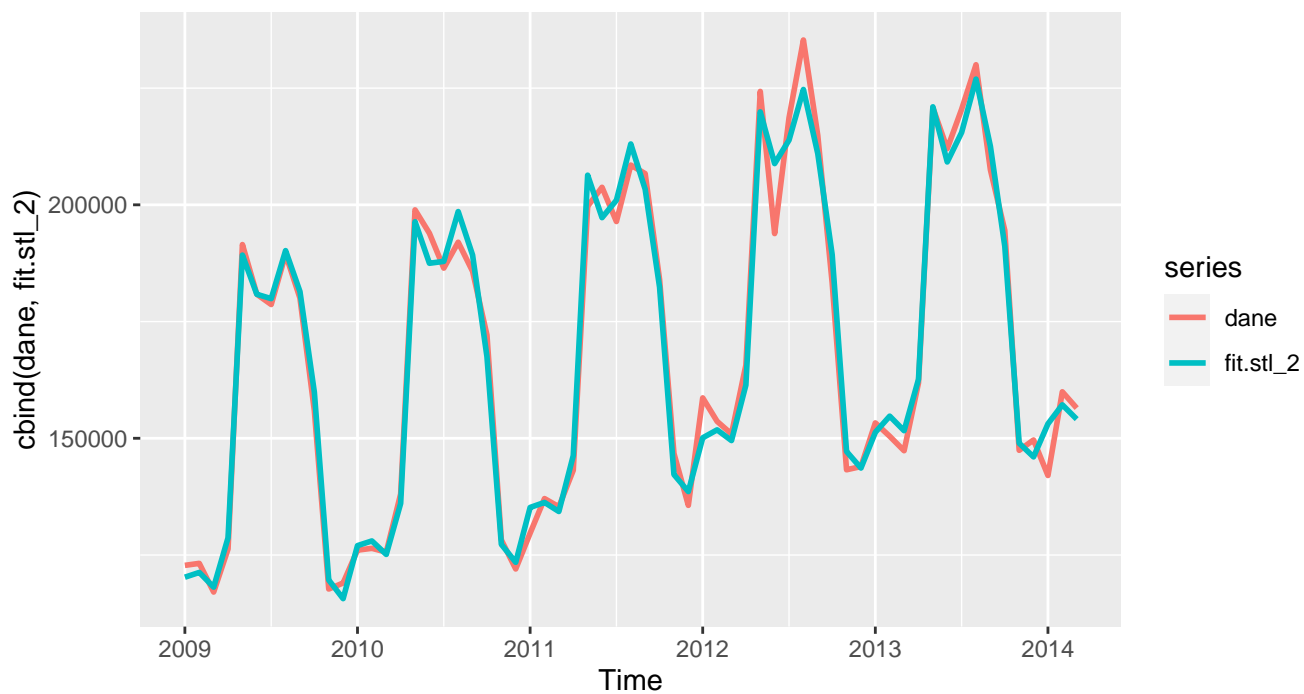
Wykres składowych zamieszczamy na rysunku niżej.





Rysunek 26: Wykres składowych: stl, model 2

Sprawdzimy następnie, jak dany model dopasował się do naszych danych.



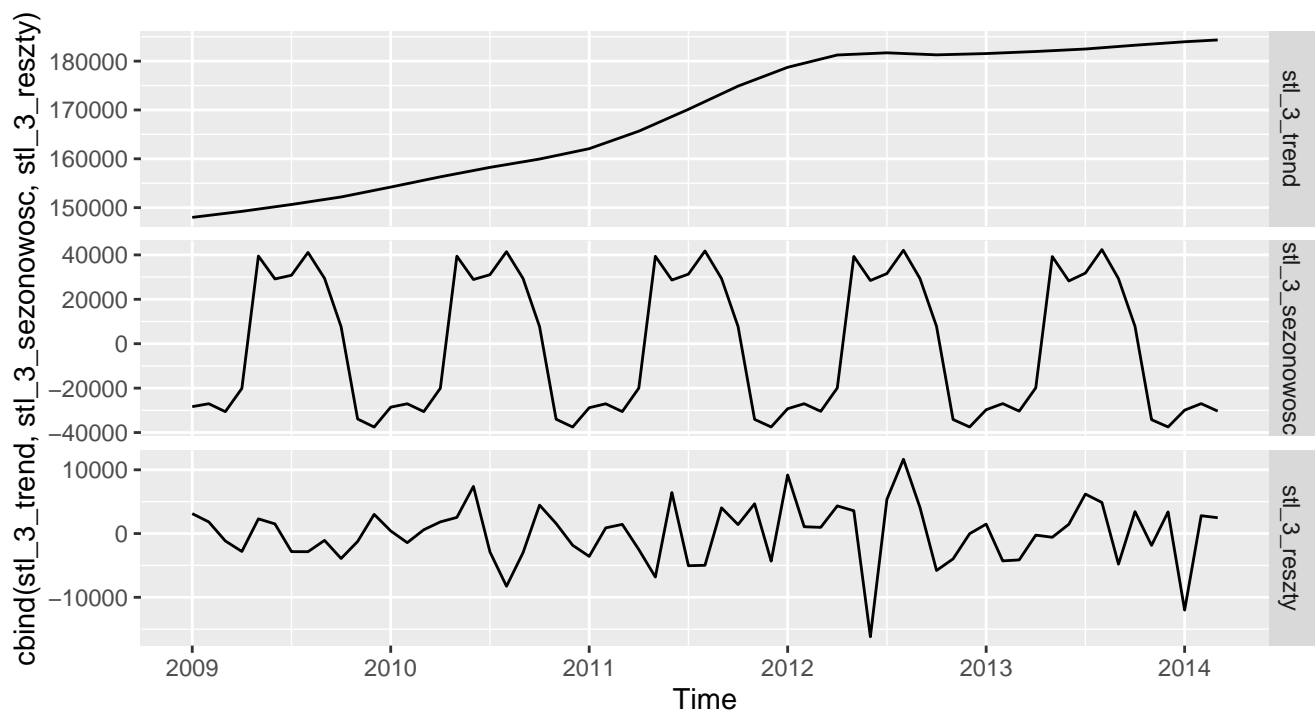
Rysunek 27: Wykres dopasowania modelu do danych: stl, model 2

Model ładnie uchwytuje ogólny trend i sezonowość, ale nadal może istnieć pewna niedoszacowana zmienność, zwłaszcza w okresach, gdzie sezonowość się zmienia.

- Model 3: s.window=13, t.window - domyślne

```
stl3 <- stl(dane, s.window=13)
stl_3_trend <- trendcycle(stl3)
stl_3_sezonowosc <- seasonal(stl3)
stl_3_reszty <- remainder(stl3)
```

Wykres składowych zamieszczamy na rysunku niżej.



Rysunek 28: Wykres składowych: stl, model 3

Sprawdzimy następnie, jak dany model dopasował się do naszych danych.



Rysunek 29: Wykres dopasowania modelu do danych: stl, model 3

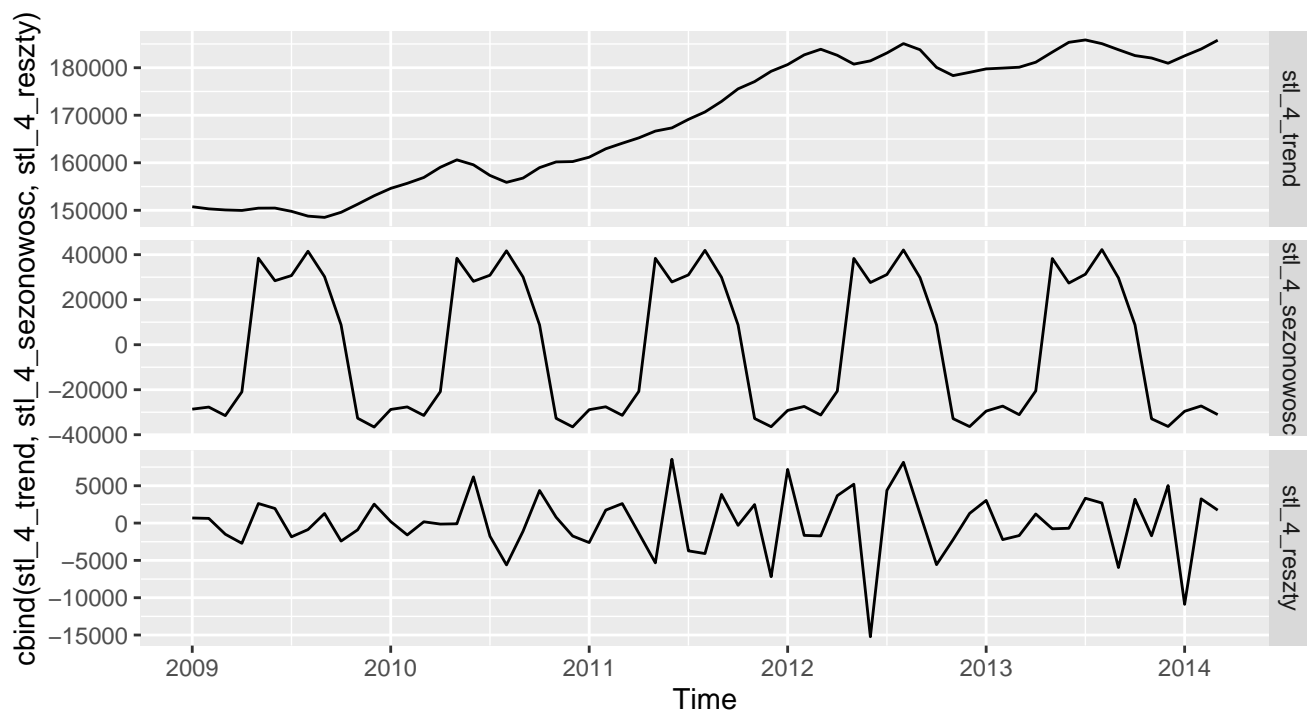
Model elastyczny dobrze radzi sobie z ogólnym trendem i bardziej zmienną sezonowością, ale może być podatny na nadmierne dopasowanie do krótkoterminowych zmian, co można zaobserwować w okresach intensywnych zmian.

- Model 4: s.window=13, t.window=7

```
stl4 <- stl(dane, s.window=13, t.window=7)

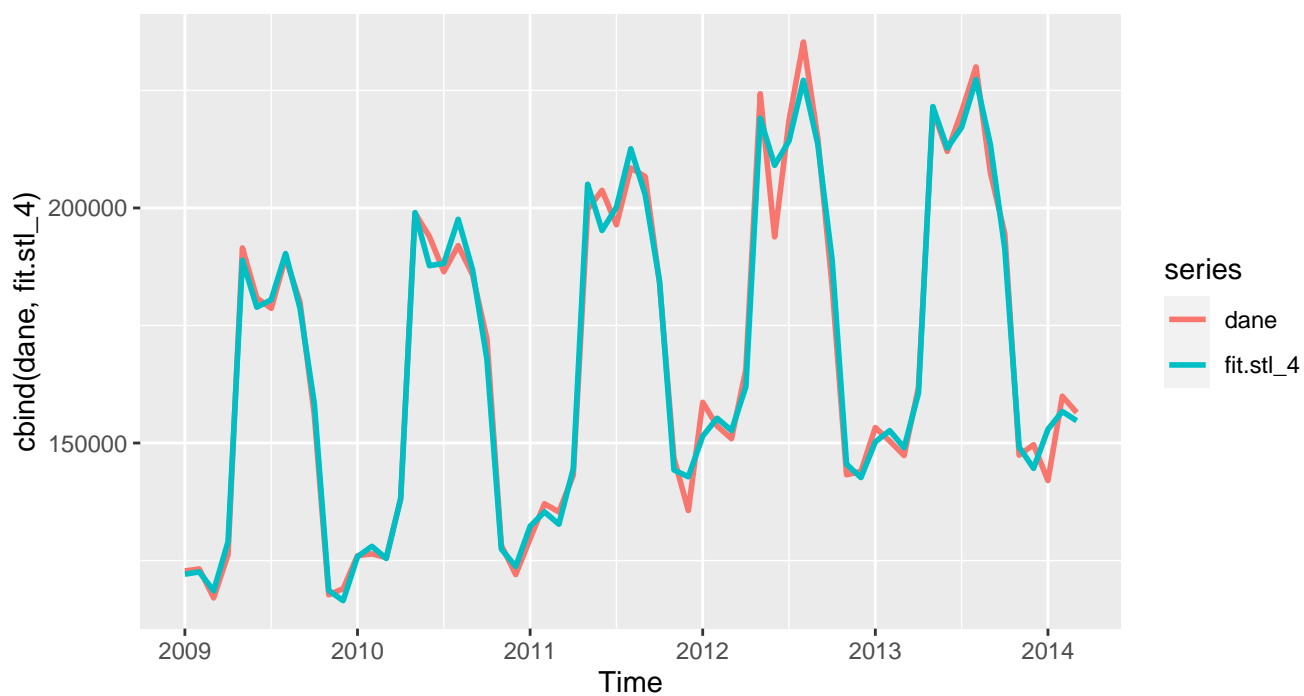
stl4_trend <- trendcycle(stl4)
stl4_sezonowosc <- seasonal(stl4)
stl4_reszty <- remainder(stl4)
```

Wykres składowych zamieszczamy na rysunku niżej.



Rysunek 30: Wykres składowych: stl, model 4

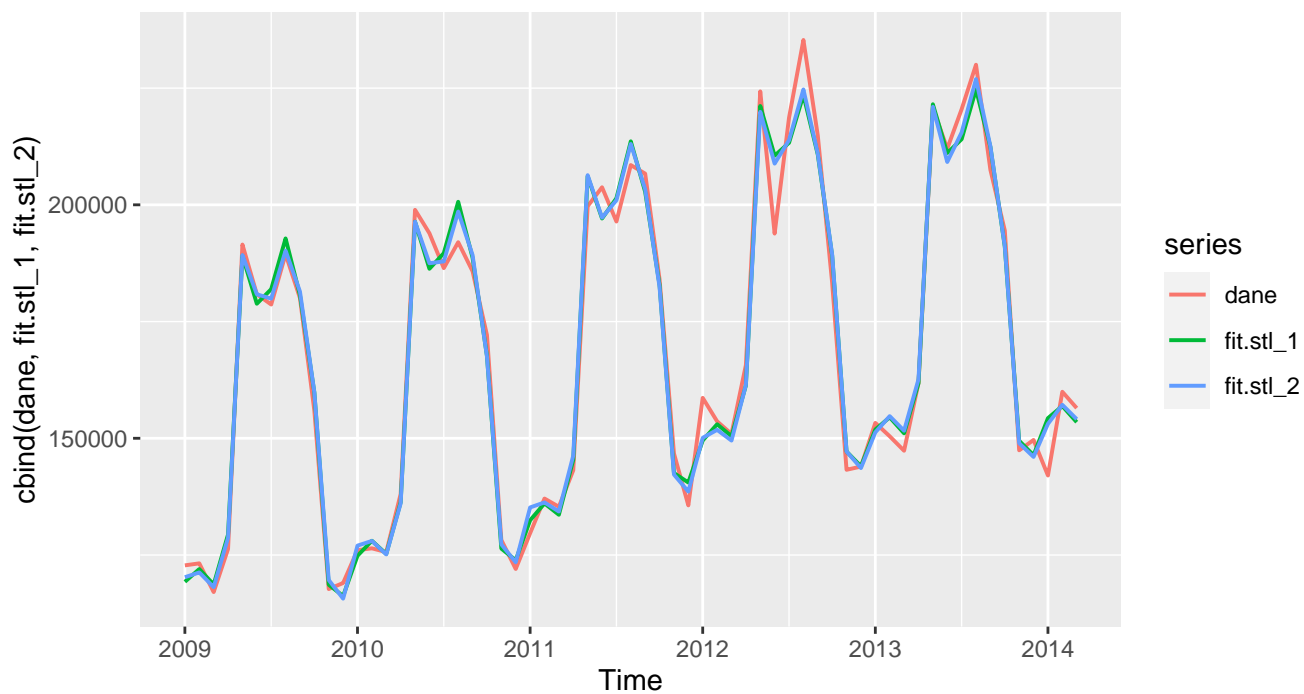
Sprawdzimy następnie, jak dany model dopasował się do naszych danych.



Rysunek 31: Wykres dopasowania modelu do danych: stl, model 4

Model radzi sobie dobrze z ogólnym trendem i bardziej zmienną sezonowością, ale wygładzanie trendu może prowadzić do niedoszacowania pewnych krótkoterminowych zmian, zwłaszcza w okresach intensywnych zmian.

Spróbujemy teraz porównać dopasowanie modeli do danych na jednym wykresie.

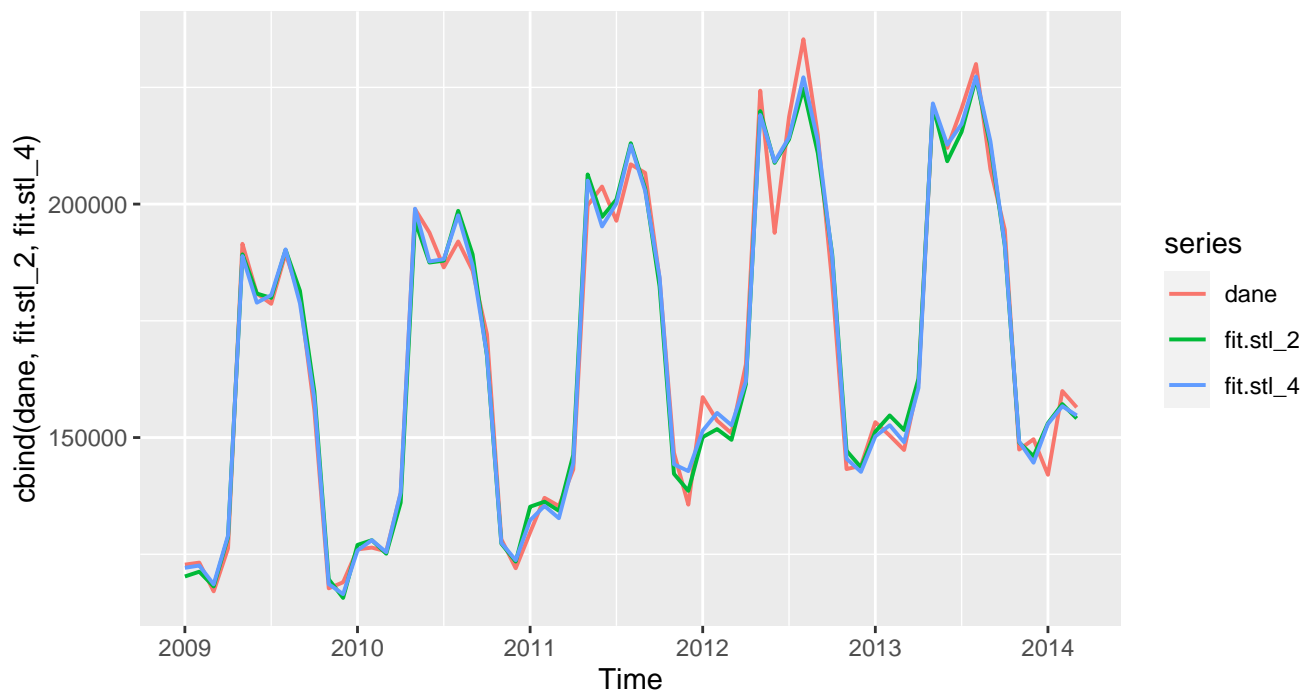


Rysunek 32: Wykres dopasowania modeli stl do danych



Rysunek 33: Wykres dopasowania modeli stl do danych

Można zauważyć, że modele 2 i 4 najbardziej się dopasowały. Porównujemy je osobno na wykresie niżej.

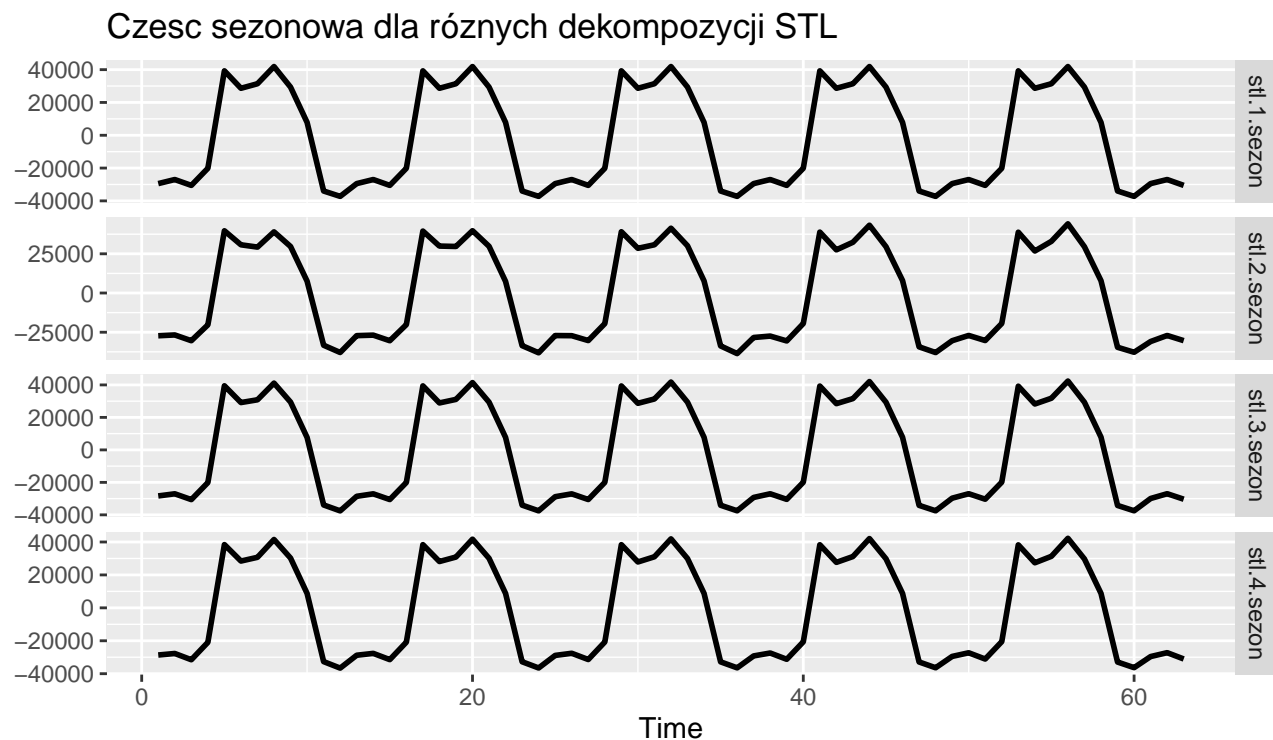


Rysunek 34: Wykres dopasowania modeli STL do danych

Model 4 (o parametrach  $s.window=13$ ,  $t.window=7$ ) najlepiej się dopasował do naszych danych i będzie reprezentował dekompozycję STL w porównaniu metod dekompozycji szeregu czasowego.

Skupimy się teraz na częściach sezonowych dla różnych dekompozycji STL





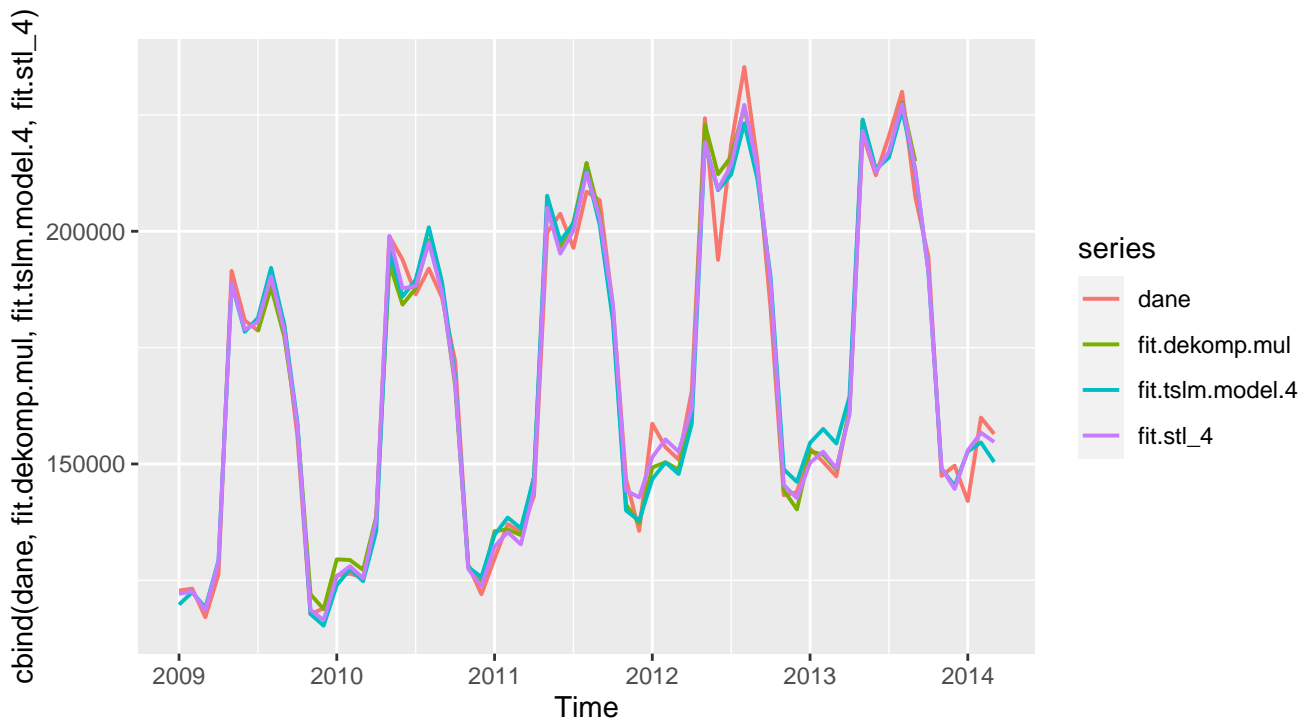
Rysunek 35: Wykres dopasowania modeli STL do danych

Widać, że nie ma istotnych różnic pomiędzy powyższymi wykresami.

#### 1.4.4 Porównanie jakości dopasowania poszczególnych modeli dekompozycji

Sporządzimy wykres dopasowania najlepszych modeli, a mianowicie:

- dekompozycja na podstawie ruchomej średniej — dekompozycja multiplikatywna;
- dekompozycja na podstawie modelu regresji — model 4 (trend sześcienny + sezonowość);
- dekompozycja STL oparta na metodzie loess — model 4 (o parametrach  $s.window=13$ ,  $t.window=7$ ).



Rysunek 36: Wykres dopasowania modeli różnych metod dekompozycji do danych

Ogólnie rzecz biorąc, każdy z tych modeli prezentuje dobrą jakość dopasowania do danych, ale najlepiej sobie poradziła metoda dekompozycji STL oparta na metodzie loess — model 4 (o parametrach  $s.window=13$ ,  $t.window=7$ ).

## 1.5 Zastosowanie transformacji Boxa–Coxa

W tym podpunkcie spróbujemy rozstrzygnąć, czy zastosowanie transformacji Boxa–Coxa prowadzi do poprawy jakości dopasowania modeli dekompozycji. Będziemy rozpatrywać kilka wartości parametru  $\lambda$ : 0 oraz za optymalne wartości za pomocą funkcji `BoxCox.lambda()`.

### 1.5.1 Dekompozycja na podstawie ruchomej średniej

Na początku wyznaczamy optymalne parametry  $\lambda$  za pomocą funkcji `BoxCox.lambda()`.

```
BoxCox.lambda(dane)
## [1] 0.4406069
BoxCox.lambda(dane, method="loglik")
## [1] 0.45
```

Ponieważ optymalne parametry, uzyskane dwiema metodami, wyszły prawie identyczne, to będziemy brać pod uwagę tylko  $\lambda = 0$  i  $\lambda = 0.45$ .

Następnie przekształcamy nasze dane.

```
dane.lambda.0 <- BoxCox(dane, lambda=0)
dane.lambda.opt_log <- BoxCox(dane, lambda=BoxCox.lambda(dane, method="loglik"))
```

Sporządzamy wykresy dopasowania dekompozycji addytywnej przez zastosowaniem transformacji Boxa-Coxa i po.

```
## Error in decompose(dane.lambda.0): 'dane.lambda.0'
## Error in decompose(dane.lambda.opt_log): 'dane.lambda.opt_log'
## Error in eval(expr, envir, enclos): 'dekomp.add.lambda.0'
## Error in eval(expr, envir, enclos): 'dekomp.add.lambda.opt_log'
## Error in InvBoxCox(fit.dekomp.add.lambda.0, lambda = 0): 'fit.dekomp.add.lambda.0'
## Error in InvBoxCox(fit.dekomp.add.lambda.opt_log, lambda = BoxCox.lambda(dane, : 'fit.dekomp.add.lambda.0'
## Error in vapply(sers, is.null, NA): 'fit.dekomp.add.lambda.0'
```

Transformacja Boxa-Coxa poprawiła dopasowanie względem dekompozycji addytywnej. Najlepsze dopasowanie wykazał parametr  $\lambda = 0.45$ , otrzymany za pomocą funkcji *BoxCox.lambda()*.

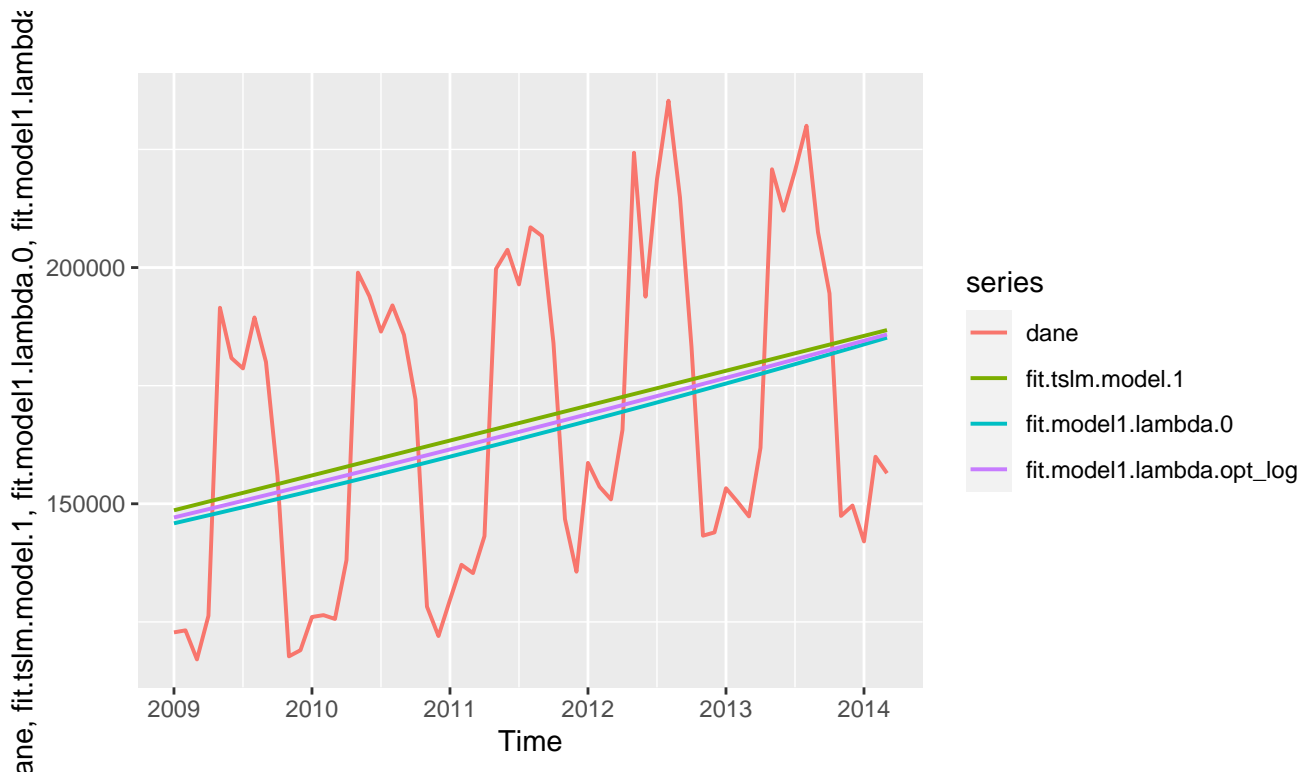
Zajmiemy się teraz porównaniem dopasowania dekompozycji multiplikatywna i dekompozycji addytywnej po transformacji Boxa-Coxa.

```
## Error in vapply(sers, is.null, NA): 'fit.dekomp.add.lambda.opt_log'
```

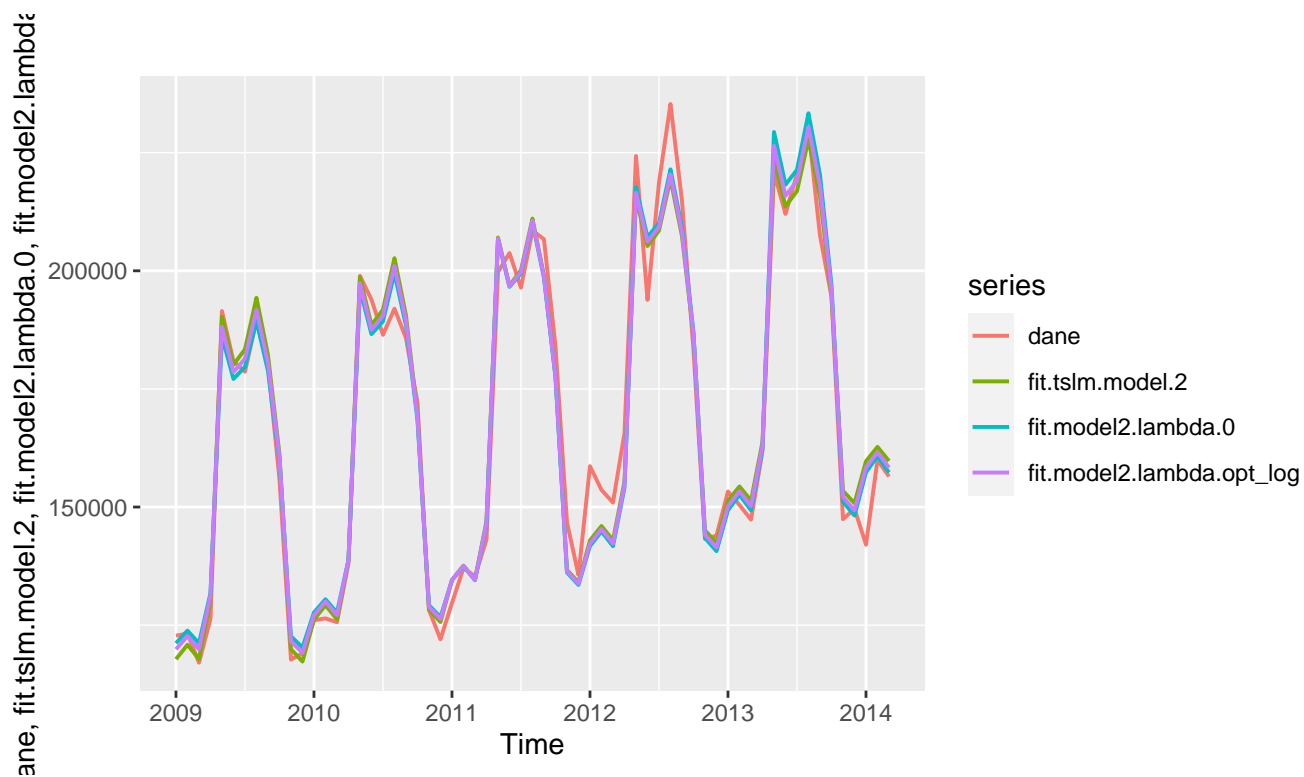
Widać, że dekompozycja addytywna po transformacji daje praktycznie takie same rezultaty jak dekompozycja multiplikatywna. co sugeruje, że dana transformacja poprawiła jakość dopasowania w przypadku dekompozycji addytywnej.

### 1.5.2 Dekompozycja na podstawie modelu regresji

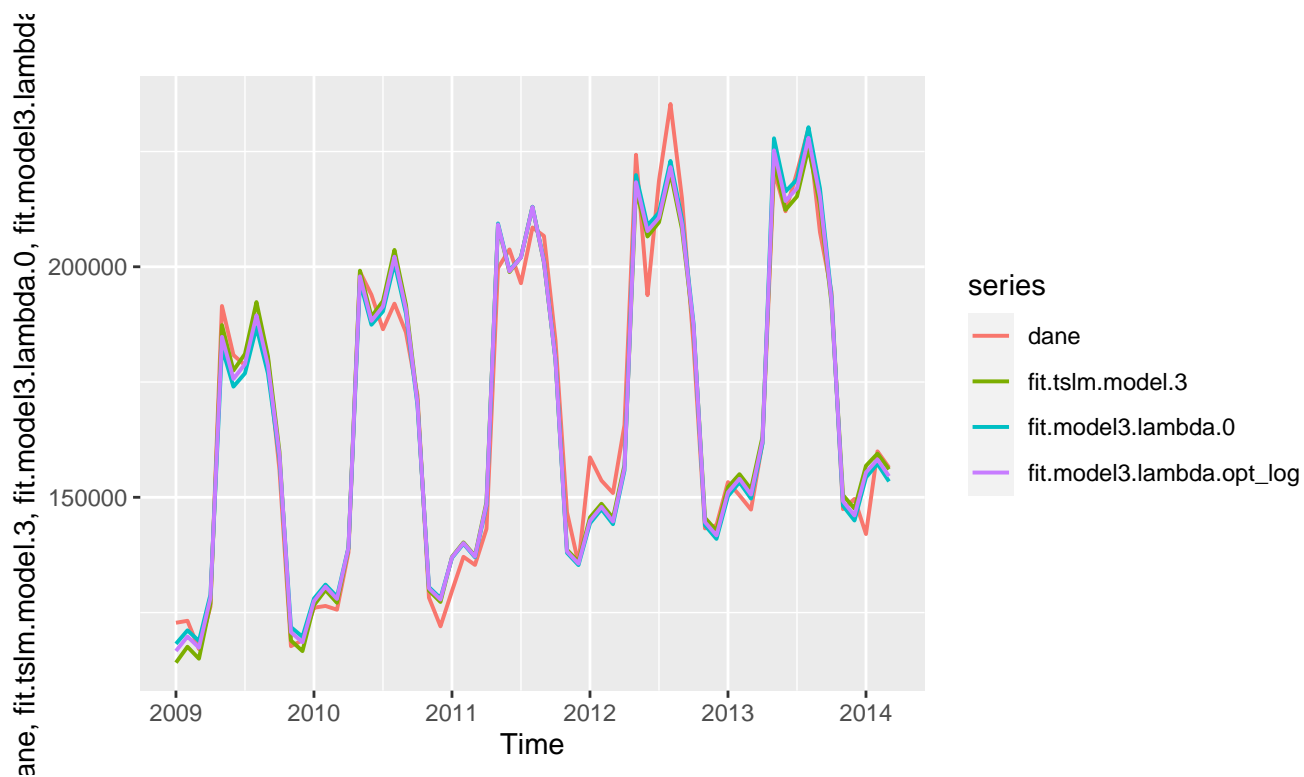
Sporządzamy wykresy dopasowania dekompozycji na podstawie modelu regresji dla pięciu modeli przed zastosowaniem transformacji Boxa-Coxa i po.



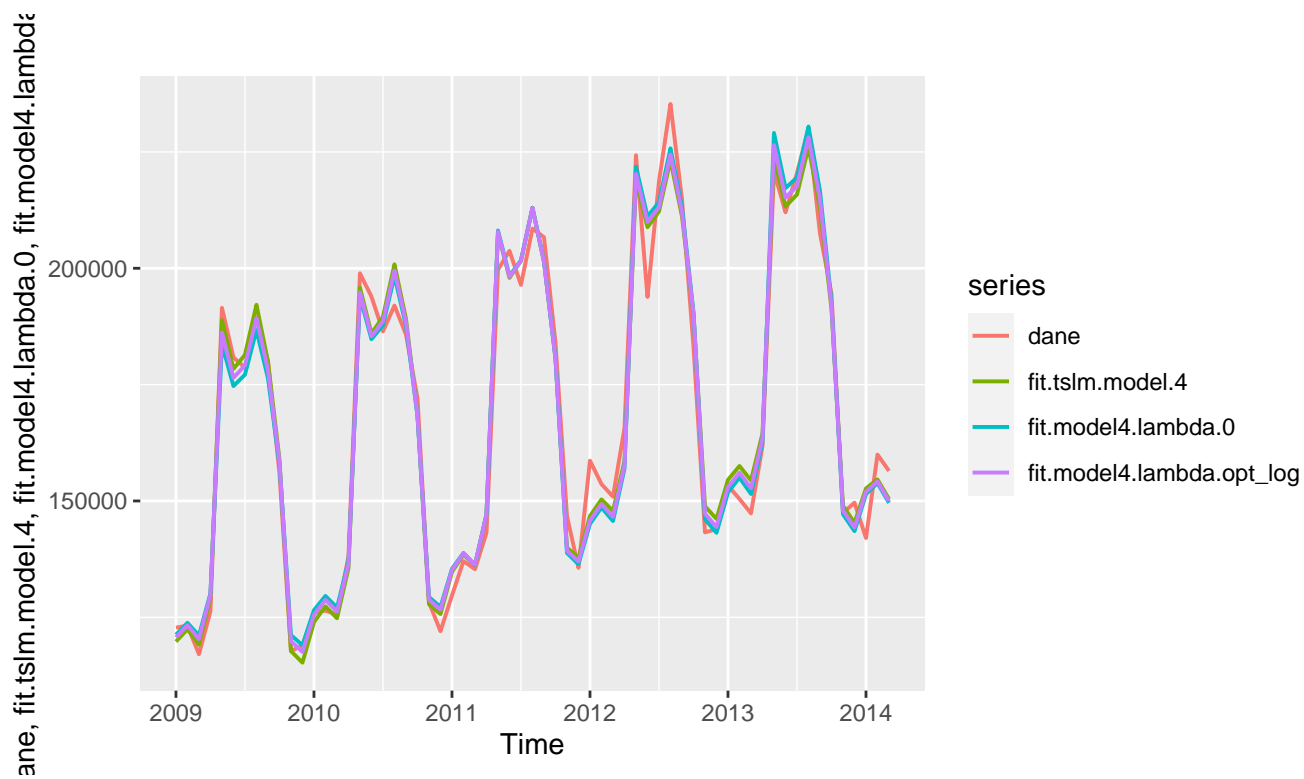
Rysunek 37: Wykres porównania dopasowania modelu 1 dekompozycji na podstawie modelu regresji przed i po transformacji Boxa-Coxa



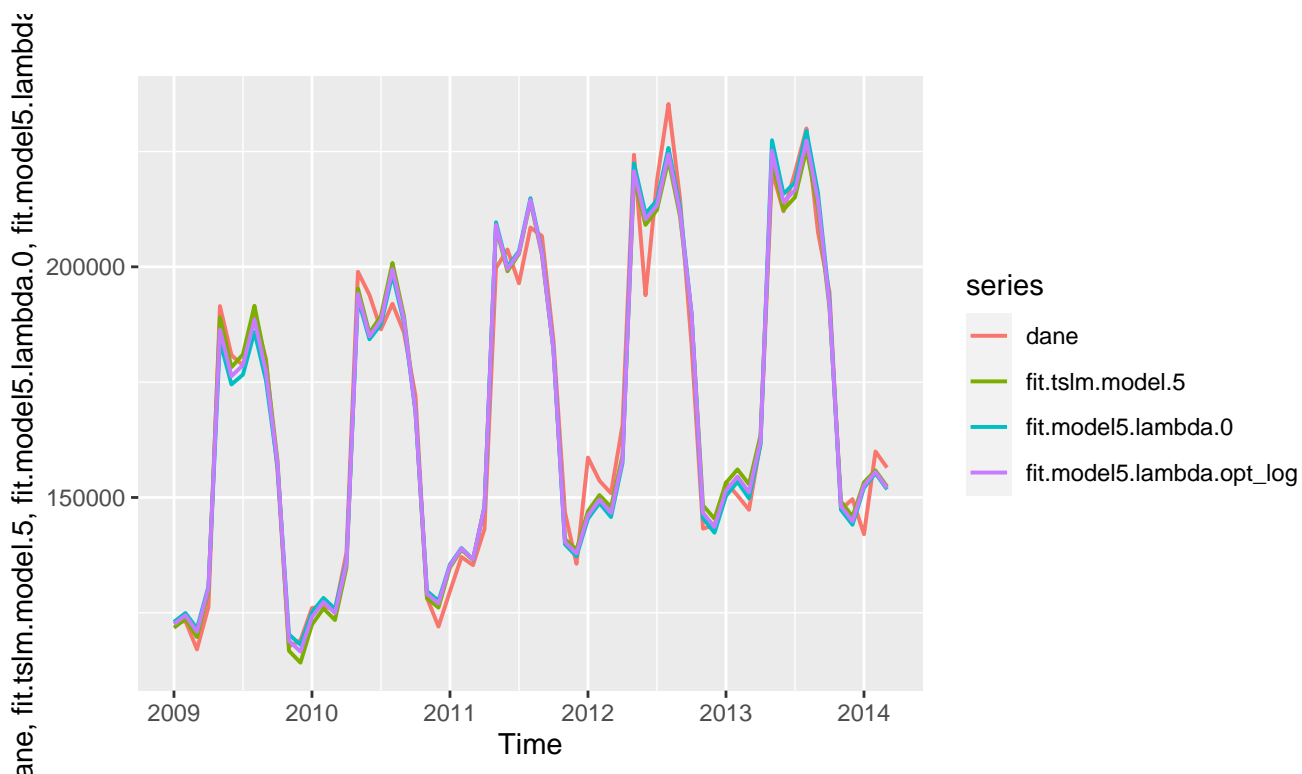
Rysunek 38: Wykres porównania dopasowania modelu 2 dekompozycji na podstawie modelu regresji przed i po transformacji Boxa-Coxa



Rysunek 39: Wykres porównania dopasowania modelu 3 dekompozycji na podstawie modelu regresji przed i po transformacji Boxa-Coxa



Rysunek 40: Wykres porównania dopasowania modelu 4 dekompozycji na podstawie modelu regresji przed i po transformacji Boxa-Coxa



Rysunek 41: Wykres porównania dopasowania modelu 5 dekompozycji na podstawie modelu regresji przed i po transformacji Boxa-Coxa

Dla modelu 1: dopasowanie jest porównywalnie złe, brak uwzględnienia sezonowości w modelu regresji jest mocno widoczny we wszystkich trzech przypadkach.

Dla modeli 2-5: transformacja Boxa-Coxa nie wpływa istotnie na jakość dopasowania.

Wyznamy szczegółowe informacje dotyczące modeli po transformacji, takie jak współczynniki regresji, wartości t-statystyki,  $R^2$  za pomocą funkcji *summary*.

- Model 1

Dla  $\lambda = 0$ .

[1] 0.1205792

Dla  $\lambda = 0.45$ .

[1] 0.1159037

Wnioski: nie możemy oceniać ulepszenia jakości dopasowania dla tego modelu ze względu na brak uwzględnienia sezonowości.

- Model 2

Dla  $\lambda = 0$ .

[1] 0.9673416

Dla  $\lambda = 0.45$ .

[1] 0.968861

- Model 3

Dla  $\lambda = 0$ .

[1] 0.9717449

Dla  $\lambda = 0.45$ .

[1] 0.9728022

- Model 4

Dla  $\lambda = 0$ .

[1] 0.9747225

Dla  $\lambda = 0.45$ .

[1] 0.976724

- Model 5

Dla  $\lambda = 0$ .

[1] 0.9756852

Dla  $\lambda = 0.45$ .

[1] 0.9775443

Wnioski dla modeli 2-5: Wartość  $R^2$  jest wyższa dla  $\lambda = 0.45$ , ale wszystkie wartości (dla modelu przed transformacją i dla  $\lambda = 0$ ) są bardzo zbliżone do siebie.

### 1.5.3 Dekompozycja STL oparta na metodzie loess

Sporządzamy wykresy dopasowania dekompozycji STL oparta na metodzie loess dla czterech modeli przed zastosowaniem transformacji Boxa-Coxa i po.

```
## Error in as.ts(x): 'dane.lambda.0'
## Error in as.ts(x): 'dane.lambda.0'
## Error in as.ts(x): 'dane.lambda.0'
## Error in as.ts(x): 'dane.lambda.0'
## Error in as.ts(x): 'dane.lambda.opt_log'
## Error in as.ts(x): 'dane.lambda.opt_log'
## Error in as.ts(x): 'dane.lambda.opt_log'
## Error in as.ts(x): 'dane.lambda.opt_log'
## Error in "mstl" %in% class(object): 'stl.1.lambda.0'
## Error in "mstl" %in% class(object): 'stl.2.lambda.0'
## Error in "mstl" %in% class(object): 'stl.3.lambda.0'
## Error in "mstl" %in% class(object): 'stl.4.lambda.0'
## Error in "mstl" %in% class(object): 'stl.1.lambda.0.45'
## Error in "mstl" %in% class(object): 'stl.2.lambda.0.45'
## Error in "mstl" %in% class(object): 'stl.3.lambda.0.45'
## Error in "mstl" %in% class(object): 'stl.4.lambda.0.45'
## Error in InvBoxCox(fit.stl.1.lambda.0, lambda = 0): 'fit.stl.1.lambda.0'
## Error in InvBoxCox(fit.stl.2.lambda.0, lambda = 0): 'fit.stl.2.lambda.0'
## Error in InvBoxCox(fit.stl.3.lambda.0, lambda = 0): 'fit.stl.3.lambda.0'
## Error in InvBoxCox(fit.stl.4.lambda.0, lambda = 0): 'fit.stl.4.lambda.0'
## Error in InvBoxCox(fit.stl.1.lambda.0.45, lambda = 0.45): 'fit.stl.1.lambda.0.45'
## Error in InvBoxCox(fit.stl.2.lambda.0.45, lambda = 0.45): 'fit.stl.2.lambda.0.45'
## Error in InvBoxCox(fit.stl.3.lambda.0.45, lambda = 0.45): 'fit.stl.3.lambda.0.45'
## Error in InvBoxCox(fit.stl.4.lambda.0.45, lambda = 0.45): 'fit.stl.4.lambda.0.45'
```

```
## Error in vapply(sers, is.null, NA): 'fit.stl.1.lambda.0'
## Error in vapply(sers, is.null, NA): 'fit.stl.2.lambda.0'
## Error in vapply(sers, is.null, NA): 'fit.stl.3.lambda.0'
## Error in vapply(sers, is.null, NA): 'fit.stl.4.lambda.0'
```

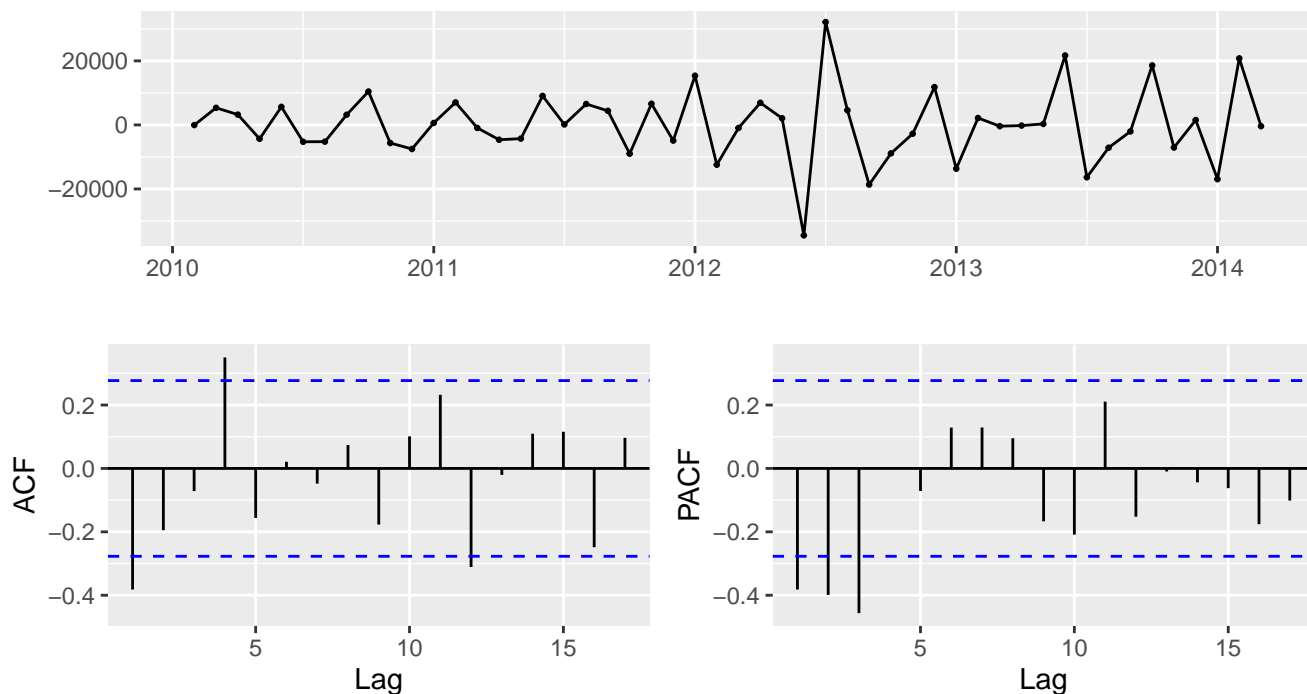


Transformacja Boxa-Coxa z parametrem  $\lambda=0$  poprawiła dopasowanie we wszystkich czterech modelach w przypadku metody dekompozycji STL.

## 1.6 Porównanie wyników eliminacji trendu i sezonowości

W tym podpunkcie porównamy wyniki eliminacji trendu i sezonowości na podstawie rozważanych metod dekompozycji z wynikami uzyskanymi poprzez odpowiednie różnicowanie danych.

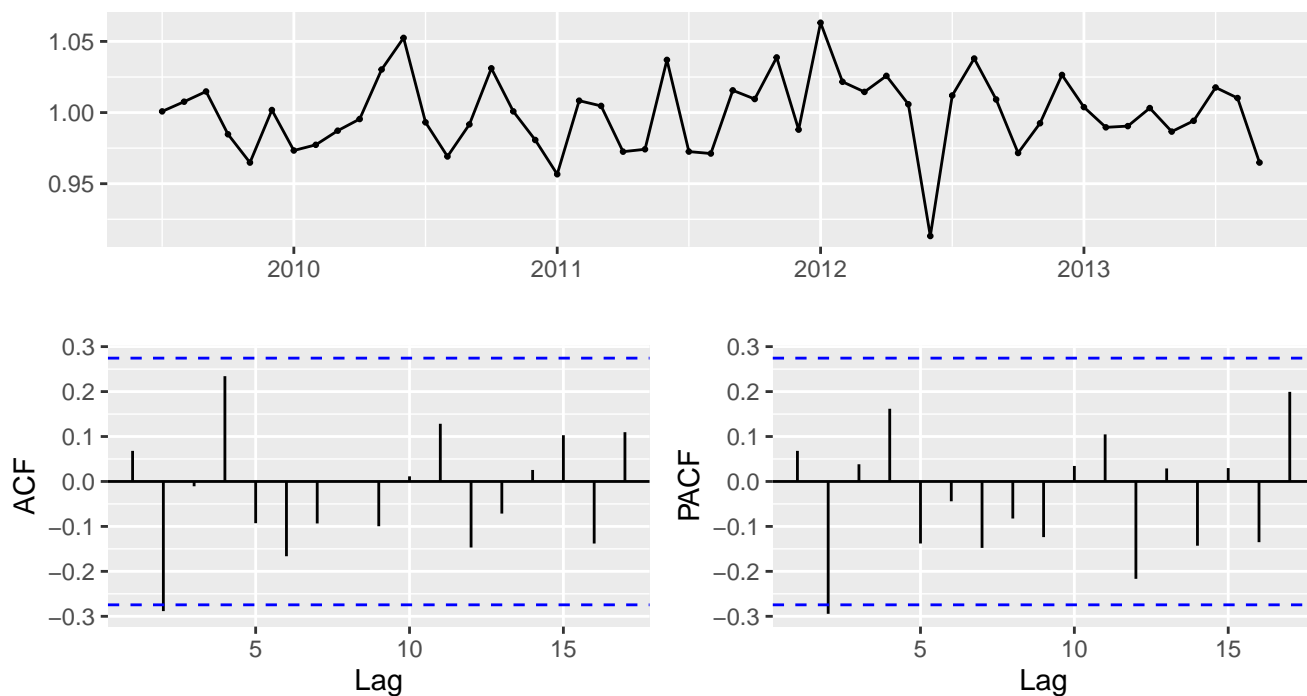
Na poniższym rysunku przedstawiono wykres otrzymanych reszt z różnicowań.



Rysunek 42: Analiza otrzymanego szeregu reszt z różnicowań

W otrzymanym szeregu reszt nie widać trendu i sezonowości, co również potwierdza to też wykres ACF. Szereg reszt ma więc stacjonarny charakter.

Na poniższym rysunku przedstawiono wykres otrzymanych reszt z dekompozycji multiplikatywnej.



Rysunek 43: Analiza wykresu reszt z dekompozycji multiplikatywnej

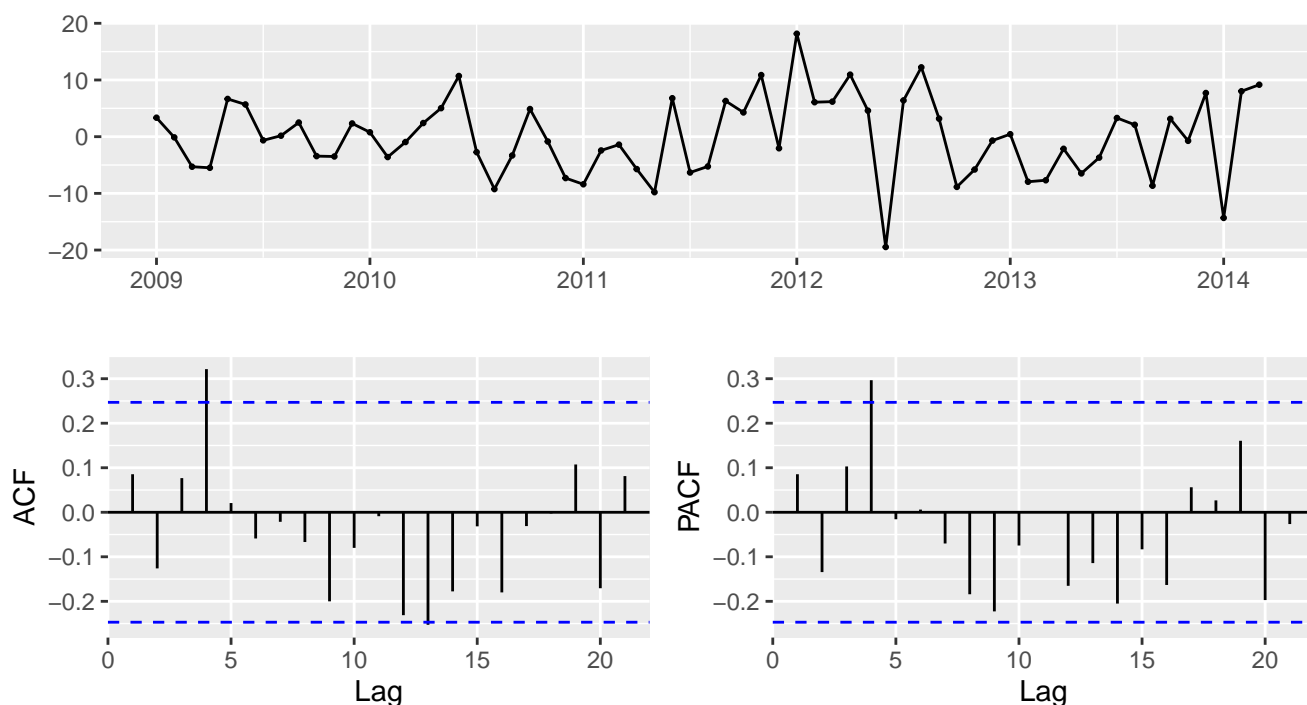
Nie widać pozostałości seoznoWości na wykresie reszt, co również wykres ACF. Szereg reszt w tym przypadku jest stacjonarny.

Na poniższym rysunku przedstawiono wykres otrzymanych reszt z dekompozycji addytywnej po transformacji Boxa-Coxa.

```
## Error in NCOL(x): 'dekomp.add.lambda.opt_log'
```

Podobnie jak dla dekompozycji multiplikatywnej.

Na poniższym rysunku przedstawiono wykres otrzymanych reszt z dekompozycji z dekompozycji opartej o model regresji + Boxa-Coxa.



Rysunek 44: Analiza wykresu reszt z dekompozycji opartej o model regresji + Boxa-Coxa

Nie widać pozostałości po trendzie i sezonowości w szeregu reszt. Zatem szereg ma charakter stacjonarny.

Na poniższym rysunku przedstawiono wykres otrzymanych reszt z dekompozycji STL + Boxa-Coxa.

```
## Error in "mstl" %in% class(object): 'stl.4.lambda.0'
```

Widać pozostałości po sezonowości np. na wykresie ACF. Szereg reszt nie jest więc stacjonarny.

## 2 Zadanie 3 - Zastosowanie wskaźników analizy technicznej w analizie szeregów czasowych

### 2.1 Opis wskaźników

Poniżej przedstawiamy opis kluczowych wskaźników analizy technicznej, zastosowanych do badania dynamicznych szeregów czasowych notowań akcji na giełdzie. Opiszemy wskaźniki takie jak: MACD, STS/SMI, Wstęga Bollingera, RSI, SMA.

#### 2.1.1 MACD

MACD (Średnia Koniunkcyjna Ruchoma) - wskaźnik ten bada zbieżności i rozbieżności średnich ruchomych, koncentrując się na różnicy między wartościami długoterminowej i krótkoterminowej średniej wykładniczej cen zamknięcia.

- **Budowa**

- Szybsza Linia (MACD) - różnica między dwiema wykładniczymi średnimi kroczącymi cen zamknięcia, zazwyczaj o okresach 12 i 26 dni lub tygodni.
- Wolniejsza Linia (Sygnał): 9-okresowa średnia wykładnicza linii MACD.

- **Interpretacja**

- Sygnały kupna i sprzedaży pojawiają się przy przecięciach linii MACD i sygnału:  
Przecięcie od dołu to sygnał kupna.  
Przecięcie od góry to sygnał sprzedaży.
- Stan wykupienia rynku jest określany, gdy linie są zbyt wysoko ponad linią zero, a stan wyprzedania, gdy spadają zbyt daleko poniżej.
- Negatywna dywergencja występuje, gdy linie MACD są powyżej zera, słabną, a ceny nadal rosną, co może sygnalizować zbliżający się szczyt cenowy.
- Pozytywna dywergencja występuje, gdy linie MACD są poniżej zera, rosną, a ceny są jeszcze niskie, co może sugerować zbliżające się dno cenowe.

### 2.1.2 STS

STS (Stochastyczny Oscylator) - jest wskaźnikiem opartym na obserwacji, że w trendach wzrostowych ceny zamknięcia zbliżają się do górnej granicy zakresu wahań, a w trendach spadkowych zbliżają się do dolnej granicy tego zakresu.

- **Budowa**

- Wykorzystuje dwie linie: %K (linia szybsza) i %D (linia wolniejsza).
- Linia %K mierzy w procentach relację ostatniej ceny zamknięcia do zakresu wahań cen w okresie, zazwyczaj przy użyciu 5-dniowego okresu.
- %D to 3-okresowa średnia ruchoma z %K

- **Interpretacja**

- Stochastyczny Oscylator oscyluje między 0 a 100, gdzie wartości powyżej 80 oznaczają wykupienie rynku, a wartości poniżej 20 oznaczają wyprzedanie.
- Sygnałami kupna i sprzedaży są przecięcia linii %K i %D.
- Wartości skrajne wyznaczone są poziomami 80 i 20.
- Negatywna dywergencja występuje, gdy linia %D jest powyżej 80, a ceny nadal rosną, co może sugerować możliwość odwrócenia trendu wzrostowego.
- Pozytywna dywergencja występuje, gdy linia %D jest poniżej 20, a ceny nadal spadają, co może sugerować możliwość odwrócenia trendu spadkowego.

### 2.1.3 SMI

SMI (Stochastic Momentum Index) - jest to wskaźnik stworzony na bazie Stochastycznego Oscylatora (STS), który pomaga identyfikować moment zmiany kierunku trendu na rynku finansowym.

- **Budowa**

- Wykorzystuje dwie linie: %K (linia szybsza) i %D (linia wolniejsza), podobnie jak STS.
- Linia %K mierzy w procentach relację różnicy między ostatnią ceną zamknięcia a najniższą ceną w określonym okresie a różnicą między najwyższą a najniższą ceną.
- %D to 3-okresowa średnia ruchoma z %K.
- Dodatkowo, SMI zawiera linię sygnałową, która jest 3-okresową średnią z %D.

- **Interpretacja**

- Stochastic Momentum Index oscyluje między -100 a +100, a poziomy powyżej +40 i poniżej -40 są uznawane za obszary wykupienia i wyprzedania.
- Sygnałami kupna i sprzedaży są przecięcia linii %K i %D, podobnie jak w przypadku STS.
- SMI jest bardziej gładki niż STS, co może pomagać w identyfikacji trendów.
- Pozytywna dywergencja występuje, gdy linia SMI rośnie, ale cena jest na niższym poziomie, co może sugerować możliwość odwrócenia trendu wzrostowego.
- Negatywna dywergencja występuje, gdy linia SMI spada, ale cena rośnie, co może sugerować możliwość odwrócenia trendu spadkowego.

### 2.1.4 Wstęga Bollingera

Ta metoda opiera się na umieszczeniu dwóch wstęg wokół średniej, oddalonych o dwa odchylenia standardowe, zwykle 20-dniowe. Odchylenie standardowe jest używane do statystycznego przedstawienia rozproszenia cen wokół wartości średniej.

- **Budowa**

- Wstęgi Bollingera składają się z trzech elementów: średniej kroczącej (zazwyczaj 20-okresowej) oraz dwóch wstęg oddalonych o dwa odchylenia standardowe od tej średniej.
- Górna wstęga często działa jako linia oporu, a dolna wstęga jako linia wsparcia.

- **Interpretacja**

- Wstęgi Bollingera mogą być traktowane jako docelowe poziomy cenowe.
- Jeśli ceny odbijają się od dolnej wstęgi i przetną średnią 20-dniową, górna wstęga staje się poziomem docelowym.
- Po spadku cen poniżej średniej 20-dniowej, dolna wstęga staje się poziomem docelowym.
- Szerokość wstęgi mierzy zmienność cen w okresie ostatnich 20 dni.
- Wzrost zmienności powoduje rozszerzenie się wstęg, a spadek zmienności skutkuje ich zwężeniem.
- Duże rozszerzenie wstęg może wskazywać na możliwy koniec trendu, podczas gdy ich zbyt duże zwężenie może sygnalizować początek nowego trendu.

### 2.1.5 RSI

To wskaźnik, który mierzy siłę i tempo zmian cen na rynku. RSI opiera się na stosunku średnich wartości wzrostu do średnich wartości spadku.

- **Budowa**

- Standardowe ustawienie tego wskaźnika wynosi 14 okresów.
- RSI umieszczany jest na skali od 0 do 100.

- **Interpretacja**

- Wartości powyżej 70 sygnalizują wykupienie rynku, a wartości poniżej 30 oznaczają wyprzedanie.
- Dywergencje pomiędzy RSI a ceną stanowią ważne ostrzeżenia.

### 2.1.6 SMA

To wskaźnik oparty na wygładzaniu danych cenowych. Oblicza się je, dodając wartości cen zamknięcia z określonego okresu i dzieląc przez liczbę tych dni.

- **Budowa** - Standardowe ustawienie tego wskaźnika wynosi 14 okresów.

- Wzór obliczeniowy:  $SMA = \frac{Cena_1 + Cena_2 + \dots + Cena_n}{n}$ , gdzie  $Cena_i$  to cena zamknięcia dla i-tego dnia, a  $n$  to liczba dni.

- **Interpretacja**

- SMA może działać jako obszar wsparcia, gdy cena jest powyżej, lub oporu, gdy jest poniżej. Przecięcie ceny z SMA może dostarczyć sygnałów dotyczących zmiany trendu.
- Kiedy cena jest powyżej SMA, sugeruje to trend wzrostowy, a kiedy poniżej, trend spadkowy.
- Przecięcia krótszego okresu SMA z dłuższym mogą generować sygnały kupna lub sprzedaży.

## 2.2 Wizualizacja wybranych wskaźników

Aby przeprowadzić analizę i wizualizować powyżej opisane wskaźniki na bieżących danych giełdowych, zainstalujemy pakiet *quantmod*. Ten pakiet umożliwia korzystanie z funkcji *getSymbols*, która została użyta do pobrania danych dotyczących spółki Microsoft od początku do końca 2022 roku z platformy Yahoo Finance.

```
# Pobieranie notowań [MSFT.raw] z yahoo za pomocą getSymbols z pakietu quantmod
MSFT.raw <- getSymbols(Symbols="MSFT", src="yahoo", auto.assign=FALSE)

# Wybór podzbioru danych z roku 2022
msft <- MSFT.raw["2022-01-01::2023-01-01"]
```

W tabeli poniżej 1 wyświetlamy fragment danych *msft*.

Tabela 1: Fragment danych spółki Microsoft dla roku 2022

	Open	High	Low	Close	Volume	Adjusted
2022-01-03	335.350	338.000	329.780	334.750	28865100	328.728
2022-01-04	334.830	335.200	326.120	329.010	32674300	323.091
2022-01-05	325.860	326.070	315.980	316.380	40054300	310.688
2022-01-06	313.150	318.700	311.490	313.880	39646100	308.233
2022-01-07	314.150	316.500	310.090	314.040	32720000	308.390
2022-01-10	309.490	314.720	304.690	314.270	44289500	308.616

Dla każdej sesji dostępne są następujące informacje:

- Date - określa datę danej sesji giełdowej.
- Open - reprezentuje cenę otwarcia akcji na początku sesji giełdowej, czyli pierwszą notowaną cenę danego dnia.
- High - przedstawia najwyższą cenę akcji osiągniętą w trakcie danej sesji.
- Low - odnosi się do najniższej ceny akcji osiągniętej w trakcie danej sesji.
- Close - określa cenę zamknięcia akcji na koniec sesji giełdowej, czyli ostatnią notowaną cenę danego dnia.
- Volume - przedstawia liczbę akcji spółki Microsoft, które zostały wymienione w trakcie danej sesji.
- Adjusted - zawiera cenę zamknięcia akcji, która została dostosowana o różne korekty, takie jak dywidendy czy podziały akcji. Cena ta lepiej odzwierciedla rzeczywistą wartość inwestycji.

Podczas analizy danych spółki Microsoft skupiamy się głównie na cenie zamknięcia. Ta wartość stanowi kluczowy punkt odniesienia do wyznaczania wskaźników technicznych, takich jak MACD, STS/SMI, Wstęga Bollingera, RSI i SMA. Poniżej na rysunku 45 przedstawiamy wykres cen zamknięcia.

```
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d1>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<8f>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<bd>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<b2>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d1>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<8f>
```











```
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>
```

**Cena zamknięcia dla spółki Microsoft w okresie 1.01.2022–1.01.2023**



Rysunek 45: Wykres cen zamknięcia dla spółki Microsoft w okresie 1.01.2022-1.01.2023

#### **Wnioski:**

Na analizowanym wykresie 45 cen zamknięcia spółki Microsoft (MSFT) w okresie od 1.01.2022 do 1.01.2023 można

zauważyć wyraźny trend malejący. Jest to widoczne poprzez regularne spadki cen, co sugeruje, że w analizowanym okresie presja sprzedających przeważała nad popytem.

### 2.2.1 MACD

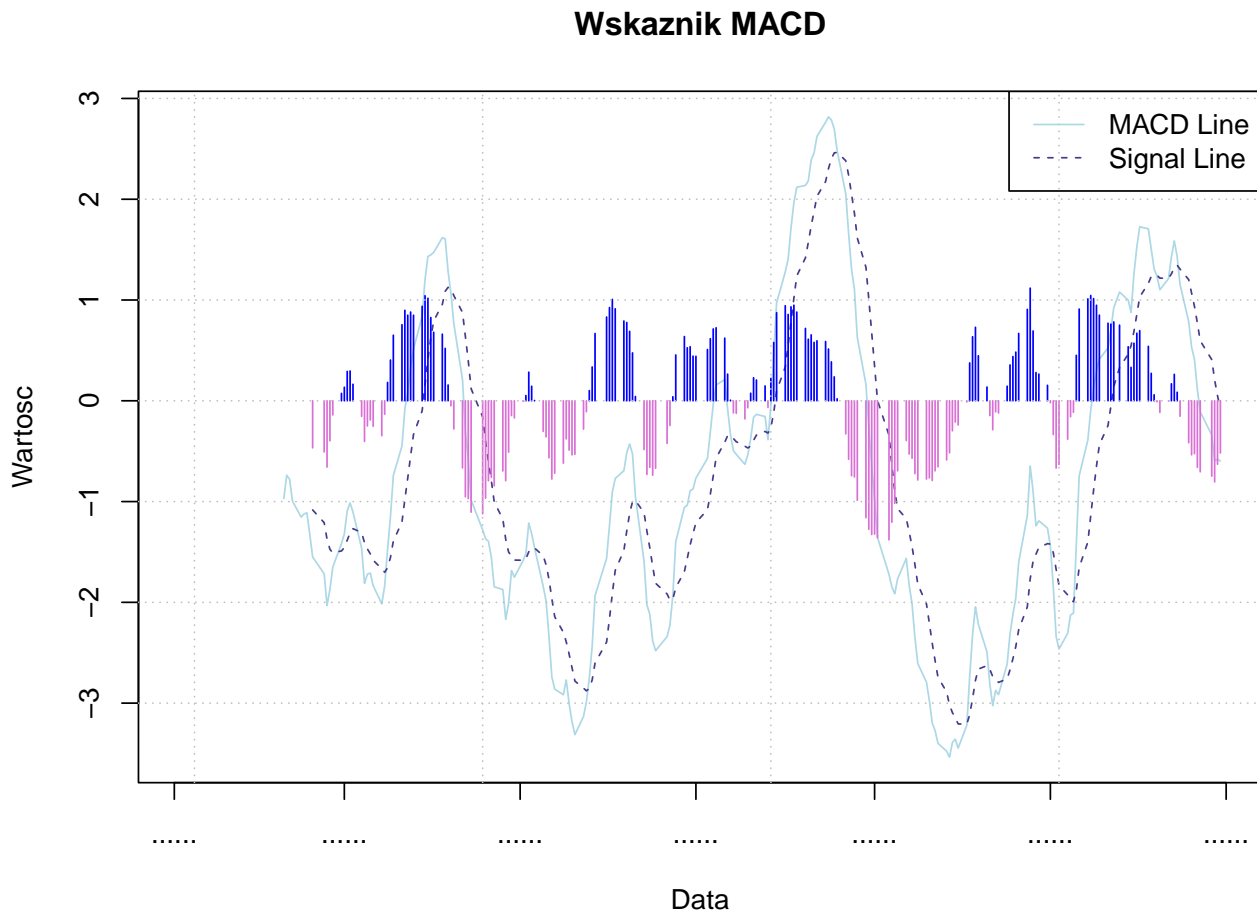
Analiza wskaźnika *MACD* na danych dotyczących ceny zamknięcia spółki Microsoft w roku 2022:

Ustawiamy parametry wskaźnika, definiując:

- okresy dla linii szybkiej (12),
- linii wolnej (26)
- linii sygnałowej (9).

```
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <8f>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bd>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b2>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <8f>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bd>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b2>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bc>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <80>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bc>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <80>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bc>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <80>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bc>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b9>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bc>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b9>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <b8>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <8e>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <bb>
## Warning in axis(side, at = z, labels = labels, ...):  '' 'mbsToSbcs':  <d0>
```





Rysunek 46: Wykres wskaźnika MACD - rok 2022

#### Wnioski:

Na podstawie analizy wykresu 46 możemy sformułować następujące wnioski:

- Pierwsze krzyżowanie, gdzie linia MACD przecina linię sygnałową z dołu, miało miejsce na początku marca, sugerując potencjalny sygnał kupna. Późniejsze krzyżowania i zmiany układu linii sygnalizowały dynamiczne zmiany w trendzie rynkowym.
- Pozytywne wartości histogramu wskazywały na okresy, gdy linia MACD była powyżej linii sygnałowej, co korelowało z okresami wzrostu cen. Ujemne wartości histogramu występowały podczas spadkowych okresów cen.
- Szczególnie istotne momenty zwiększonej odległości między liniami takie jak np. w marcu świadczą o silnych trendach rynkowych.
- Dłuższe okresy przewagi linii MACD nad sygnałową, zauważone zwłaszcza od połowy sierpnia do końca października, wskazywały na dominację trendu wzrostowego. Potwierdzeniem tego okresu dominacji było także osiągnięcie najwyższych wartości MACD i sygnału, prawie 3 dla MACD i około 2.5 dla sygnału w połowie sierpnia. Natomiast przewaga linii sygnałowej nad MACD była widoczna zwłaszcza w okresie od listopada do końca grudnia. W tym czasie wartości MACD pozostawały poniżej wartości sygnałowej, co sygnalizowało trend spadkowy na rynku. Najdłuższą przewagę sygnałową nad MACD odnotowano w pierwszej połowie listopada.

- Zmiany kierunku histogramu potwierdzają zmiany w trendzie, na przykład, gdy przechodził z dodatnich na ujemne wartości.

### 2.2.2 SMI

Analiza wskaźnika *SMI* na danych dotyczących ceny zamknięcia spółki Microsoft w roku 2022:

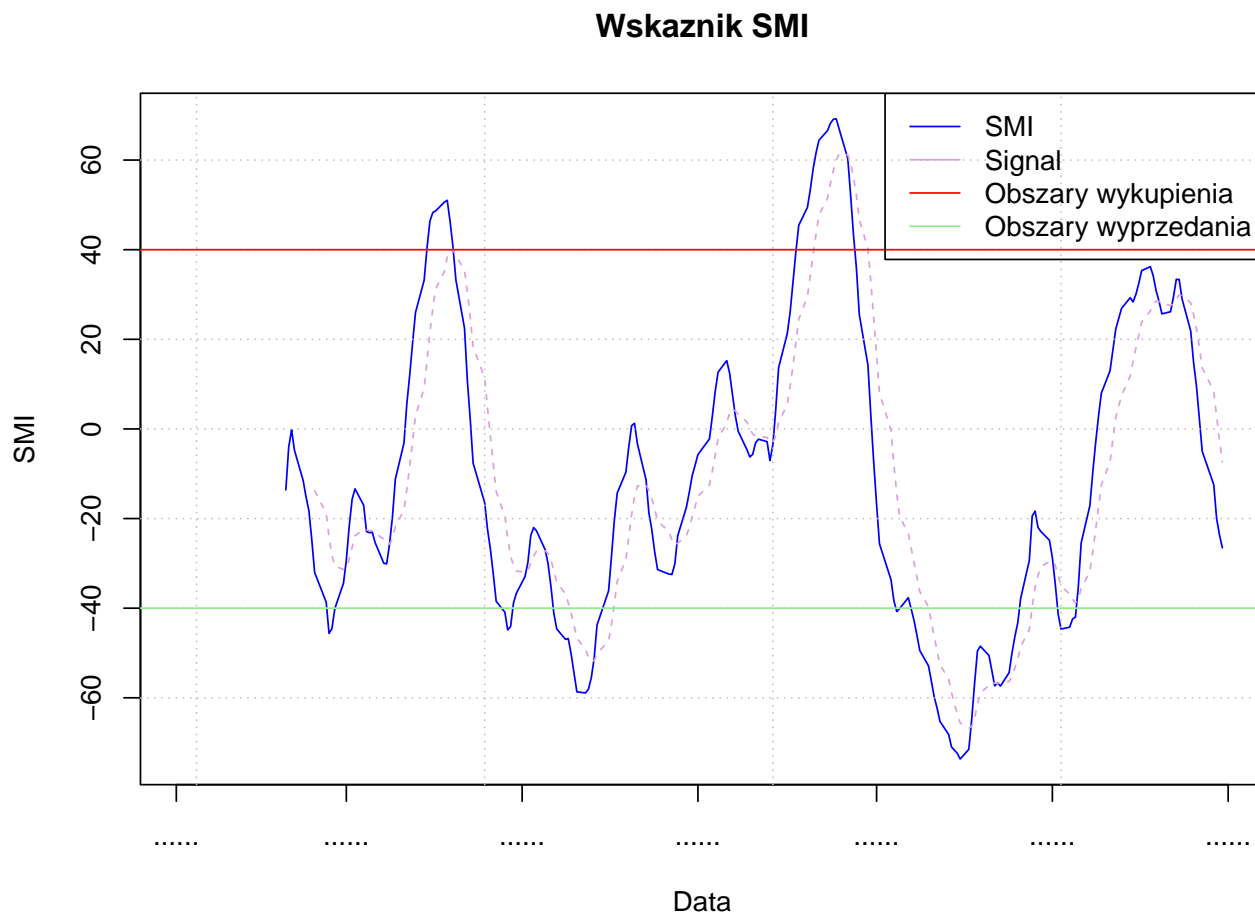
Ustawiamy parametry wskaźnika, definiując:

- liczbę okresów (13)
- liczbę okresów używanych do obliczenia linii %K (2)
- liczbę okresów używanych do obliczenia linii %D (25)
- liczbę okresów używanych do obliczenia linii sygnałowej (9).

```
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <8f>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bd>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b2>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <8f>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bd>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b2>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <80>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <80>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b9>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b9>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <b8>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <8e>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbsToSbcs': <bb>
```







Rysunek 47: Wykres dla wskaźnika SMI - rok 2022

#### Wnioski:

Na podstawie analizy wykresu 47 możemy sformułować następujące wnioski:

- Wartości SMI powyżej +40, zwłaszcza w kwietniu i sierpniu, sugerują obszary potencjalnego wykupienia (przesunięcie w stronę wzrostową). Wartości SMI poniżej -40, szczególnie od końca lutego do marca, od końca kwietnia do połowy czerwca i od września do listopada, wskazują obszary potencjalnego wyprzedania (przesunięcie w stronę spadkową).
- Przekięcia linii SMI i sygnałowej nad obszarami wykupienia mogą sugerować potencjalne sygnały sprzedaży, zwłaszcza w kwietniu i sierpniu. Przekięcia linii SMI i sygnałowej pod obszarami wyprzedania mogą sugerować potencjalne sygnały kupna, zwłaszcza od kwietnia do maja i od września do października.
- Wartości SMI poniżej -60, zwłaszcza w okresie października, sugerują obszary potencjalnie silnego wyprzedania na rynku. Warto zwrócić uwagę na okres od października do listopada, gdzie obie linie SMI i sygnałowa są poniżej -60, co może sygnalizować okres trwałego trendu spadkowego.

#### 2.2.3 Wstęga Bollingera

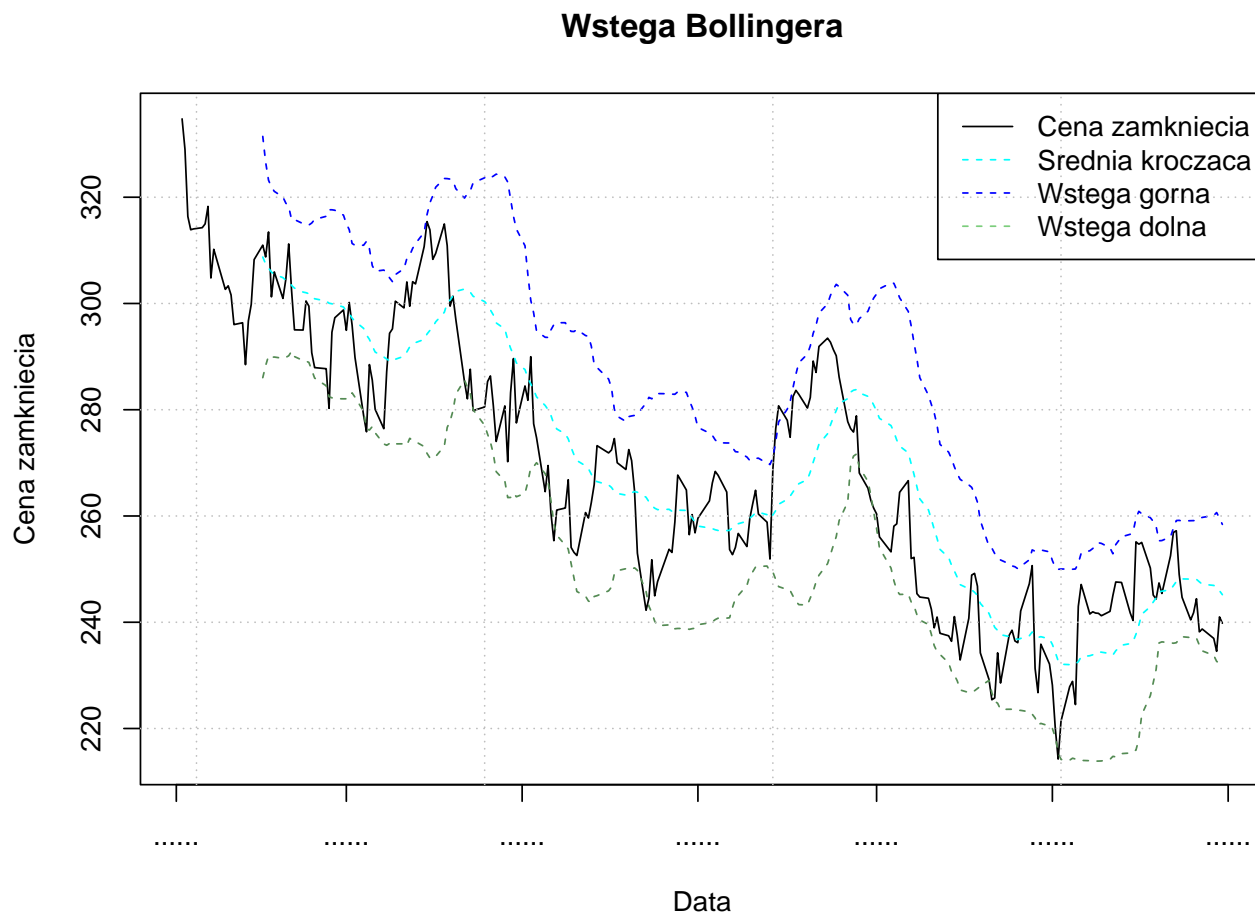
Analiza wskaźnika *Wstęga Bollingera* na danych dotyczących ceny zamknięcia spółki Microsoft w roku 2022:

Ustawiamy parametry wskaźnika, definiując:

- okres dla średniej kroczącej (20).
- współczynnik mnożenia odchylenia standardowego (2).







Rysunek 48: Wykres dla wskaźnika Wstęga Bollingera - rok 2022

#### Wnioski:

Na podstawie analizy wykresu 48 możemy sformułować następujące wnioski:

- Pod koniec lipca cena zamknięcia znajduje się ponad wstęgą górną, co może sugerować sygnał do sprzedaży. Jednocześnie jest znacznie ponad średnią kroczącą, co może wskazywać na silny trend wzrostowy. Na początku marca i pod koniec czerwca cena zamknięcia znajduje się poniżej wstęgi dolnej, co może sugerować istotny dołek cenowy i okazję do kupna. Jednocześnie w obu przypadkach jest znacznie poniżej średniej kroczącej, co może sygnalizować potencjalny trend spadkowy.  
sygnalizowało istotny dołek cenowy i okazję do kupna.
- Największa odległość między wstęgą górną a dolną występuje pod koniec czerwca oraz w okresie od początku listopada do grudnia, co wskazuje na okres dużej zmienności cenowej. To może być interpretowane jako potencjalny okres znaczącego ruchu cenowego.
- Wyżej wymienione sytuacje, gdy cena zamknięcia jest znacznie ponad (lipiec) lub poniżej (marzec, czerwiec) wstęg Bollingera, jednocześnie z relatywnie dużym odchyleniem od średniej kroczącej, mogą potwierdzać trend rynkowy.

### 2.2.4 RSI

Analiza wskaźnika *RSI* na danych dotyczących ceny zamknięcia spółki Microsoft w roku 2022:

Ustawiamy parametry wskaźnika, definiując:

- okres (14)
- poziom wykupienia (70) jest to poziom, powyżej którego *RSI* sygnalizuje, że dany instrument finansowy może być przekupiony. Na wykresie jest oznaczony niebieską linią.
- poziom wyprzedania (30) jest to poziom, poniżej którego *RSI* sygnalizuje, że dany instrument finansowy może być przeceniony. Na wykresie jest oznaczony fioletową linią.

```
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d1>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<8f>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<bd>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<b2>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d1>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<8f>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<bd>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<b2>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d1>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<84>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<b2>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<bc>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<b0>
```

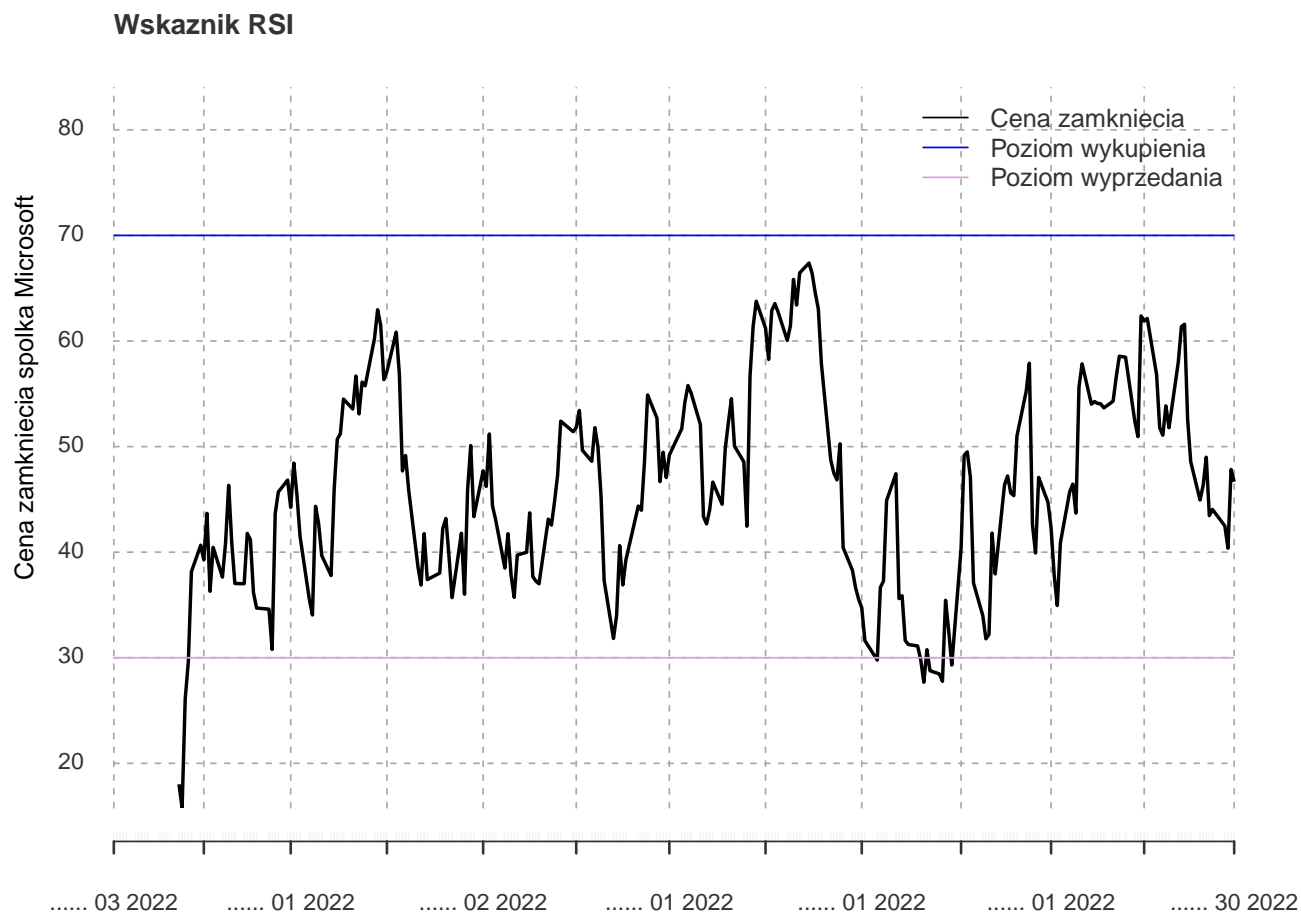












Rysunek 49: Wykres dla wskaźnika RSI - rok 2022

#### Wnioski:

Na podstawie analizy wykresu 49 możemy sformułować następujące wnioski:

- W okresie od początku roku do końca stycznia oraz od połowy do końca sierpnia, wartość RSI spadła poniżej poziomu 30, co sugeruje obszar wyprzedania. Wartości RSI poniżej 30 mogą sugerować potencjalne punkty zakupu, zwłaszcza gdy cena znajduje się na niższym poziomie.
- W żadnym momencie nie zaobserwowano przekroczenia poziomu 70, co oznacza brak sygnałów sugerujących obszar wykupienia. Brak przekroczenia poziomu 70 może sygnalizować unikanie obszarów przekupienia i czekanie na ewentualne sygnały sprzedaży.
- Brak utrzymywania się *RSI* w obszarze wyprzedania i wykupienia może świadczyć o ostrożności inwestorów przed nadmiernym wyprzedaniem/przekupieniem rynku.

#### 2.2.5 SMA

Analiza wskaźnika *SMA* na danych dotyczących ceny zamknięcia spółki Microsoft w roku 2022:

Ustawiamy parametry wskaźnika, definiując:

- okres krótkoterminowy (14) - granatowa linia,
- okres długoterminowy (50) - niebieska linia.









```

## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>

```

### Wskaźnik SMA



Rysunek 50: Wykres dla wskaźnika SMA - rok 2022

### Wnioski:

Na podstawie analizy wykresu 50 możemy sformułować następujące wnioski:

- Na wykresie obserwujemy przecięcie krótkoterminowej i długoterminowej SMA pod koniec kwietnia, co może sugerować potencjalny sygnał sprzedaży. Sytuacja powtarza się na początku sierpnia.
- Pod koniec lipca obserwujemy przecięcie krótkoterminowej SMA i długoterminowej SMA (krótkoterminowa jest nad długoterminową), co mogło być sygnałem kupna.
- Wzorce przecięć średnich kroczących sugerują dominację trendu spadkowego w analizowanym okresie. Inwestorzy mogą interpretować to jako potencjalny sygnał, że istnieje presja sprzedaży na rynku



## 2.3 Wpływ wyboru parametrów dla poszczególnych wskaźników na otrzymane wnioski

### 2.3.1 MACD

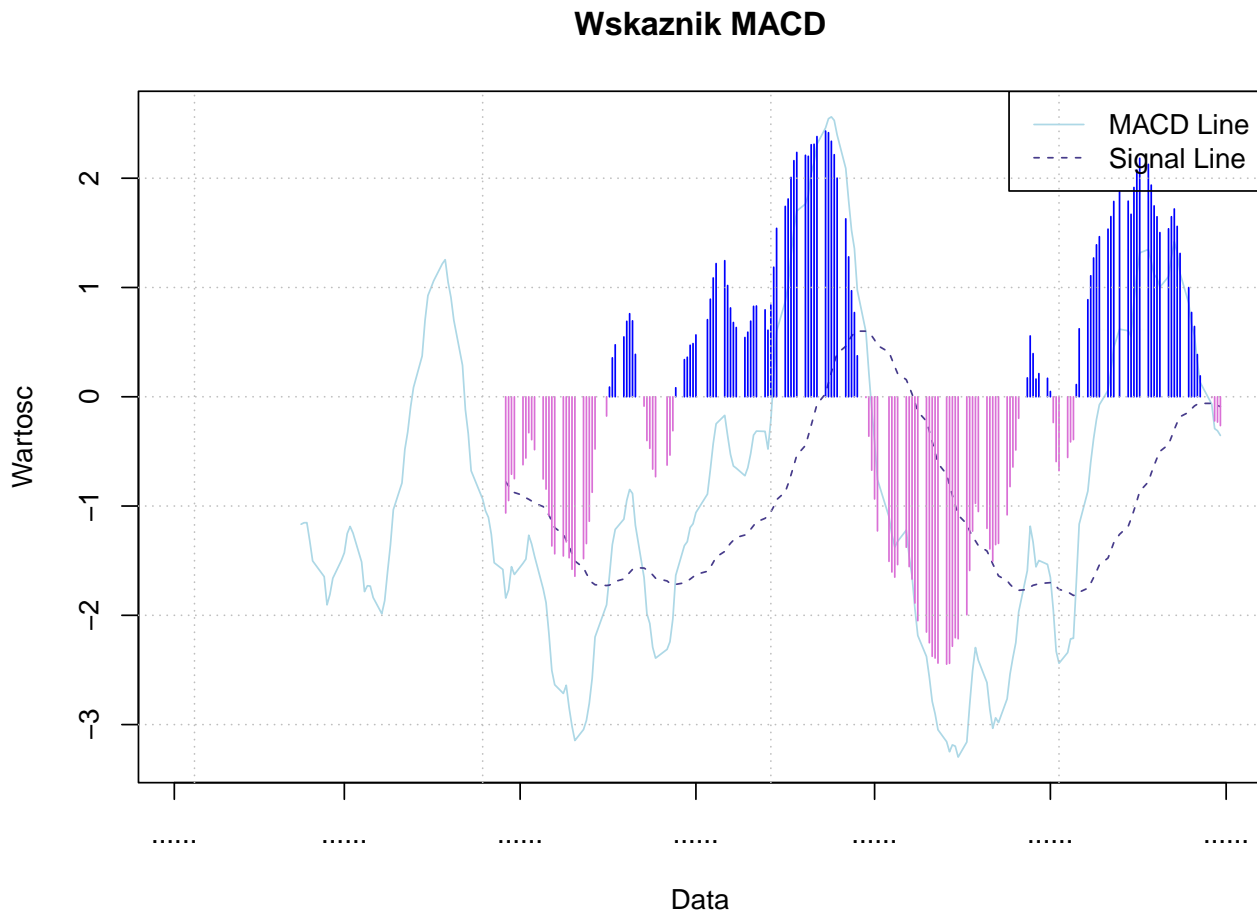
Zmienimy parametry wskaźnika *MACD* i porównamy wyniki do wyników otrzymanych na rysunku 46.

Ustawiamy parametry wskaźnika, definiując:

- okresy dla linii szybkiej (15),
- linii wolnej (30)
- linii sygnałowej (50).

```
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <8f>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bd>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b2>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <8f>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bd>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b2>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <80>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <80>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b9>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bc>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b9>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b8>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d1>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <8e>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <bb>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <d0>
## Warning in axis(side, at = z, labels = labels, ...): '' 'mbcsToSbcs': <b8>
```





Rysunek 51: Wykres wskaźnika MACD(15,30,50) - rok 2022

#### Wnioski:

- Zwiększenie okresów dla linii szybkiej, wolnej i sygnałowej skutkowało wyższymi wartościami histogramu na rysunku 51. To może wynikać z większej różnicy między wartościami MACD Line a Signal Line, co z kolei podkreśla dynamikę zmian między tymi liniami.
- Zwiększenie okresu dla linii sygnałowej (z 9 do 50) sprawiło, że linia ta stała się mniej dopasowana. To może prowadzić do dłuższych okresów utrzymania pozycji na rynku, ale jednocześnie wprowadzać opóźnienia w identyfikacji zmian trendu.
- W przypadku rysunku 51 można zauważyć, że histogram rzadziej przecina poziom zero w porównaniu do rysunku 46. Mniejsza liczba przejść przez zero może sugerować, że sygnały generowane przez wskaźnik są bardziej długoterminowe, a zmiany trendu są identyfikowane z większym opóźnieniem.

#### 2.3.2 SMI

Zmienimy parametry wskaźnika *SMI* i porównamy wyniki do wyników otrzymanych na rysunku 47.

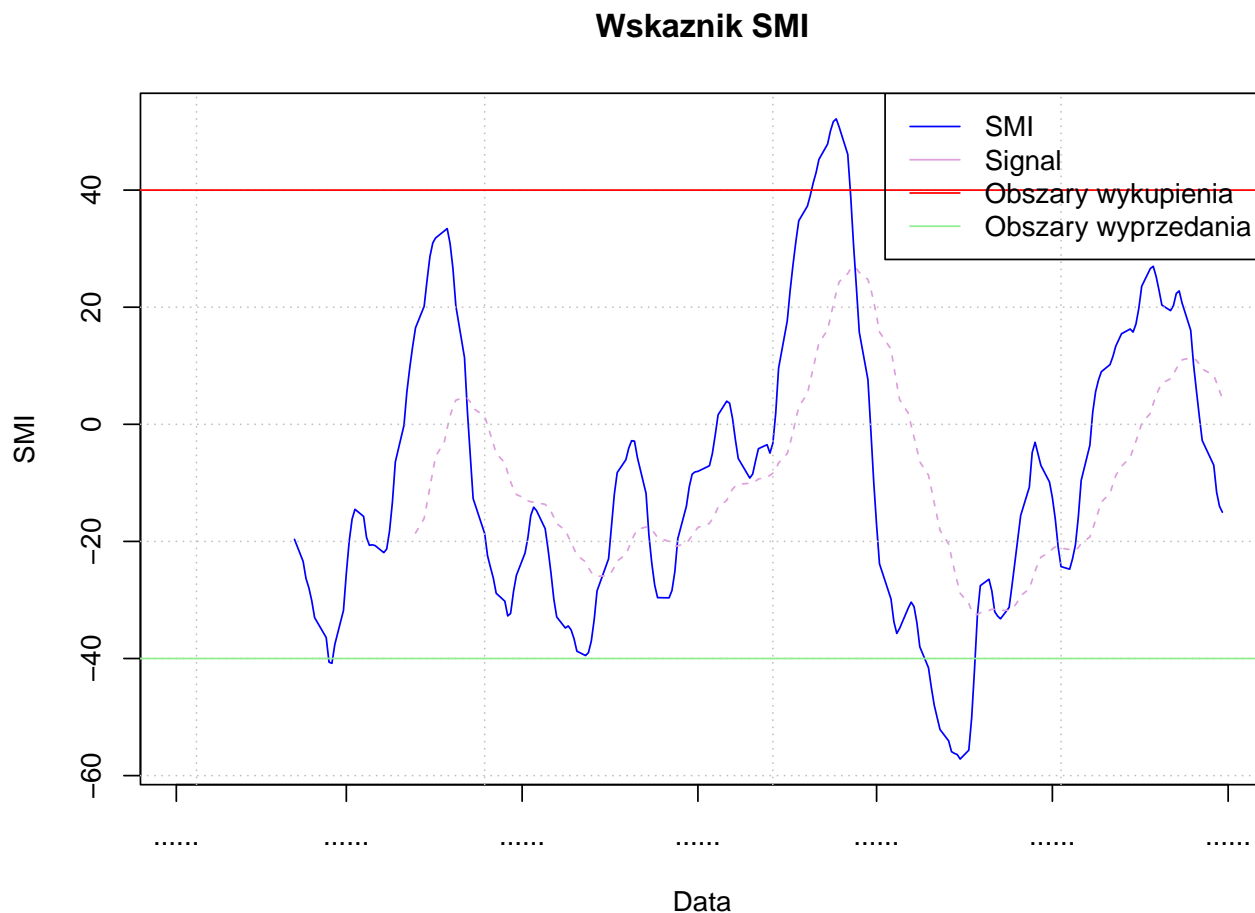
Ustawiamy parametry wskaźnika, definiując:

- liczbę okresów (5)
- liczbę okresów używanych do obliczenia linii %K (10)

- liczbę okresów używanych do obliczenia linii %D (20)
- liczbę okresów używanych do obliczenia linii sygnałowej (30).

[illegible]





Rysunek 52: Wykres dla wskaźnika SMI(9,10,20,30) - rok 2022

#### Wnioski:

- Zmiana okresu parametrów doprowadziła do zmniejszenia ilości obszarów wyprzedania i wykupienia.
- Zmiana parametrów wpłynęła również na dopasowanie linii sygnałowej.
- Na wykresie 52 linia sygnałowa rzeczywiście nie przekracza wartości +40 i -40, co wskazuje na bardziej umiarkowane i mniej skrajne sygnały w porównaniu do wykresu 47. Może to oznaczać, że zmienione parametry bardziej skupiają się na sygnałach o mniejszym ryzyku, pomijając skrajne warunki rynkowe.

#### 2.3.3 Wstęga Bollingera

Zmienimy parametry wskaźnika *Wstęga Bollingera* i porównamy wyniki do wyników otrzymanych na rysunku ??  
Ustawiamy parametry wskaźnika, definiując:

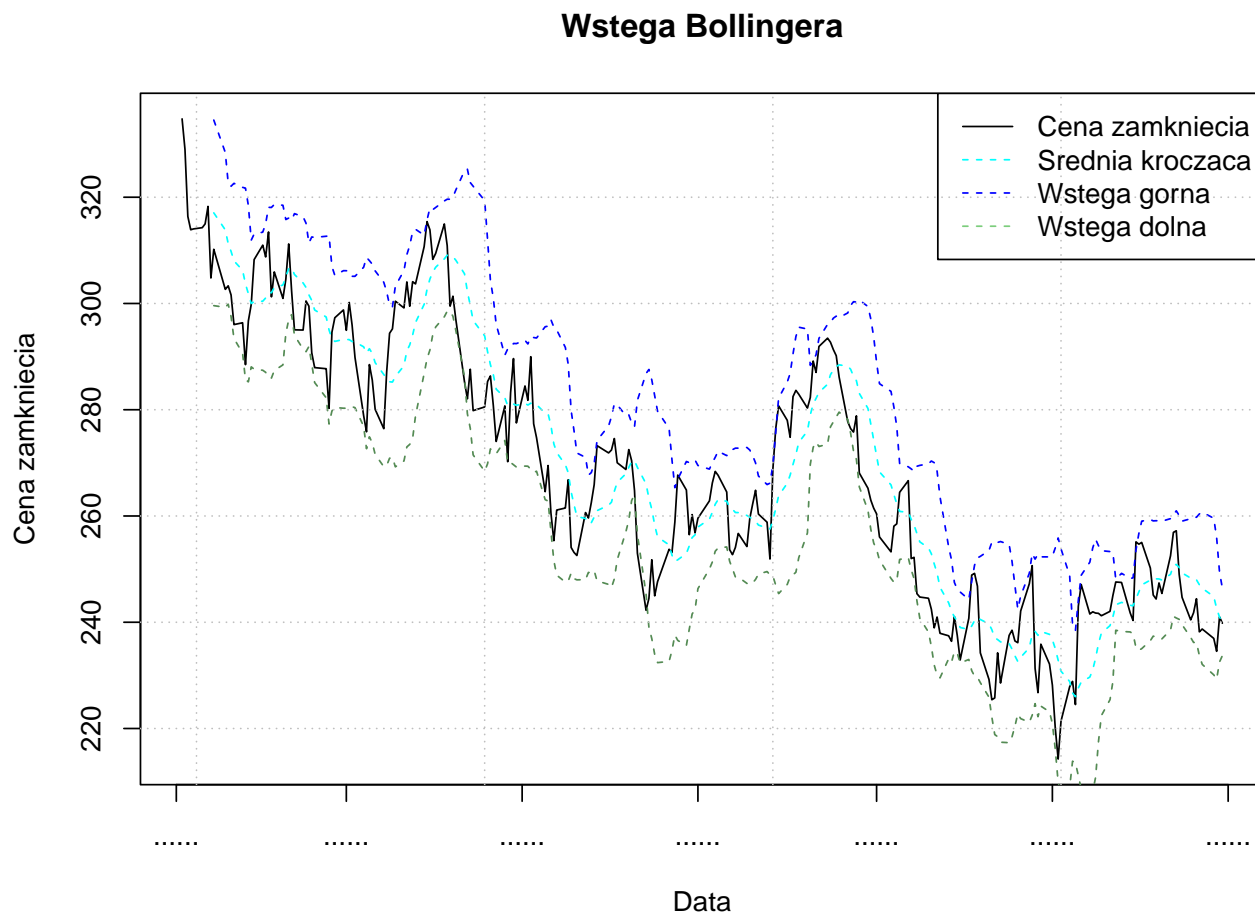
- okres dla średniej kroczącej (10).
- współczynnik mnożenia odchylenia standardowego (2).

```
## Warning in axis(side, at = z, labels = labels, ...):  ''  'mbsToSbcs':  <d1>
## Warning in axis(side, at = z, labels = labels, ...):  ''  'mbsToSbcs':  <8f>
## Warning in axis(side, at = z, labels = labels, ...):  ''  'mbsToSbcs':  <d0>
```









Rysunek 53: Wykres dla wskaźnika Wstęga Bollingera - rok 2022

#### Wnioski:

- Linia średniej kroczącej i wstęgi górna/dolna są bardziej zbliżone do ceny zamknięcia, co skutkuje częstszymi zmianami sygnałów.
- Na wykresie 53 odległość między wstęgą górną a dolną jest mniejsza w porównaniu do wykresu 48. To oznacza, że zmienione parametry sprawiły, że wstęgi są bardziej zbliżone do siebie, co sugeruje mniejszą zmienność cenową w analizowanym okresie.

#### 2.3.4 RSI

Zmienimy parametry wskaźnika *RSI* i porównamy wyniki do wyników otrzymanych na rysunku 49.

Ustawiamy parametry wskaźnika, definiując:

- okres (5)
- poziom wykupienia (70) jest to poziom, powyżej którego *RSI* sygnalizuje, że dany instrument finansowy może być przekupiony. Na wykresie jest oznaczony niebieską linią.
- poziom wyprzedania (30) jest to poziom, poniżej którego *RSI* sygnalizuje, że dany instrument finansowy może być przeceniony. Na wykresie jest oznaczony fioletową linią.





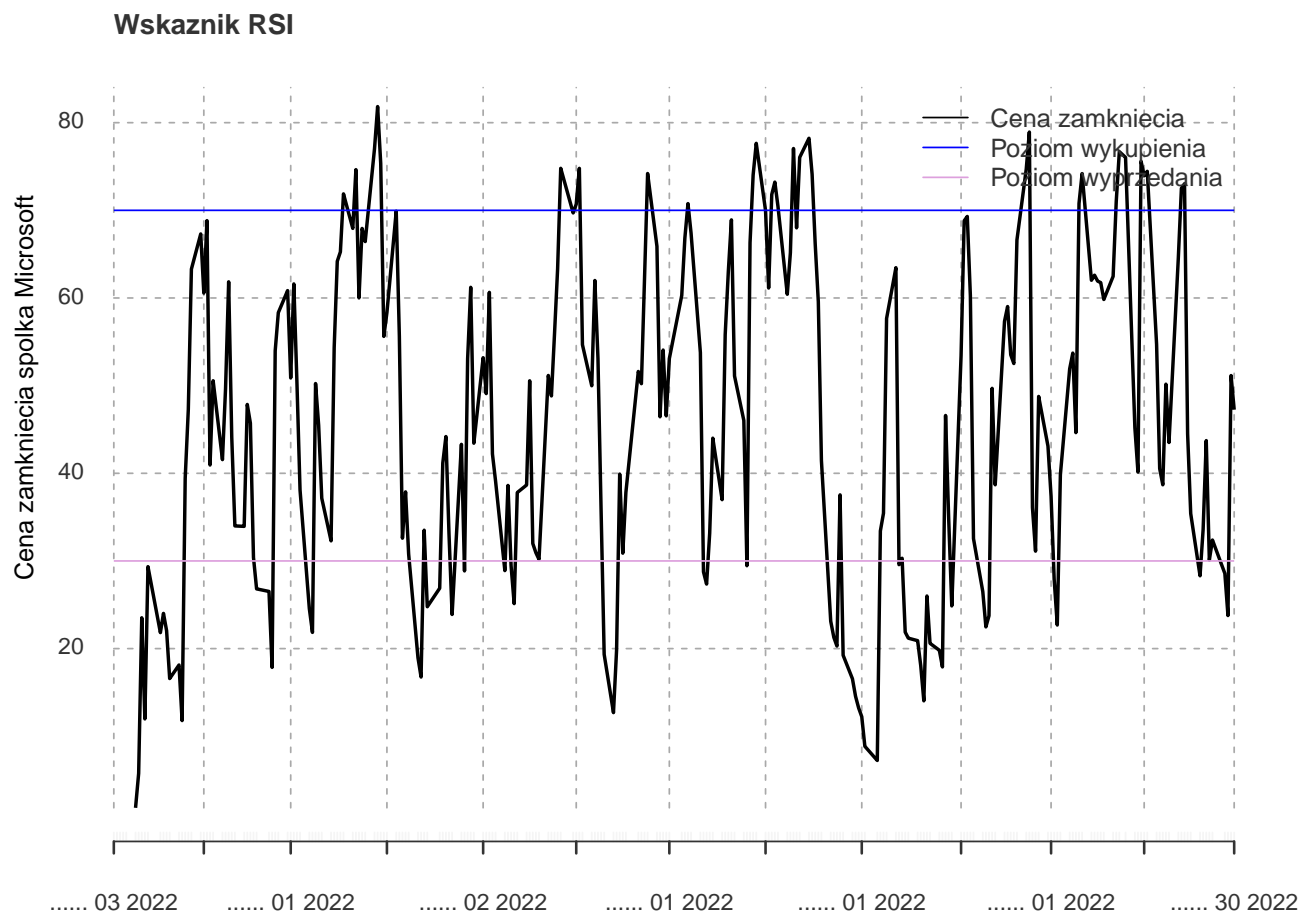




```

## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>

```



Rysunek 54: Wykres dla wskaźnika RSI(5) - rok 2022

#### Wnioski:

- Zmiana okresu  $RSI$  na 5 skutkuje bardziej dynamicznymi oscylacjami wskaźnika wokół poziomów 30 i 70.
- W przypadku  $RSI.5$  obserwujemy więcej przekroczeń poziomu 30, czyli częstsze obszary potencjalnego wyprzedania. Jednocześnie, liczba przekroczeń poziomu 70 znacznie wzrasta, stąd mamy częstsze obszary potencjalnego wykupienia.
- Brak utrzymania się  $RSI.5$  w obszarze wyprzedania i wykupienia na dłuższe okresy czasu świadczy o większej zmienności i krótszych trendach na rynku.

#### 2.3.5 SMA

Zmienimy parametry wskaźnika  $SMA$  i porównamy wyniki do wyników otrzymanych na rysunku 50.

Ustawiamy parametry wskaźnika, definiując:

- okres krótkoterminowy (5) - granatowa linia,
- okres długoterminowy (30) - niebieska linia.

```
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 03 2022' 'mbcsToSbcs
<d1>
```











```

## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 01 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b4>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<b5>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<d0>
## Warning in axis(1, at = xcoords[axt], labels = names(axt), las = theme$las, : ' 30 2022' 'mbcsToSbcs
<ba>

```

### Wskaźnik SMA



Rysunek 55: Wykres dla wskaźnika SMA(14,15) - rok 2022

### Wniosk:

- W przypadku zmienionych parametrów, krótkoterminowa linia jak i długoterminowa bardziej reagują na nagłe zmiany cen w porównaniu do poprzednich parametrów 50. Obserwujemy bardziej dynamiczną reakcję na zmienność cenową.
- Przecięcia krótkoterminowej i długoterminowej SMA są częstsze w przypadku krótszych okresów, stąd mamy więcej sygnałów handlowych.

## Literatura

- [1] Murphy, J. J. (1999). *Analiza Techniczna Rynków Finansowych*. WIG-Press, Warszawa. ISBN 83-904905-4-8.