Zadanie 2 - PWC

Aleksander Mackiewicz-Kubiak

2024-11-29

Pakiety

```r
library(tidyr)
library(gamlss)
library(dplyr)
library(fitdistrplus)
library(usefun)
library(quantmod)
library(scales)
library(copula)
library(psych)
options(scipen = 999)
```

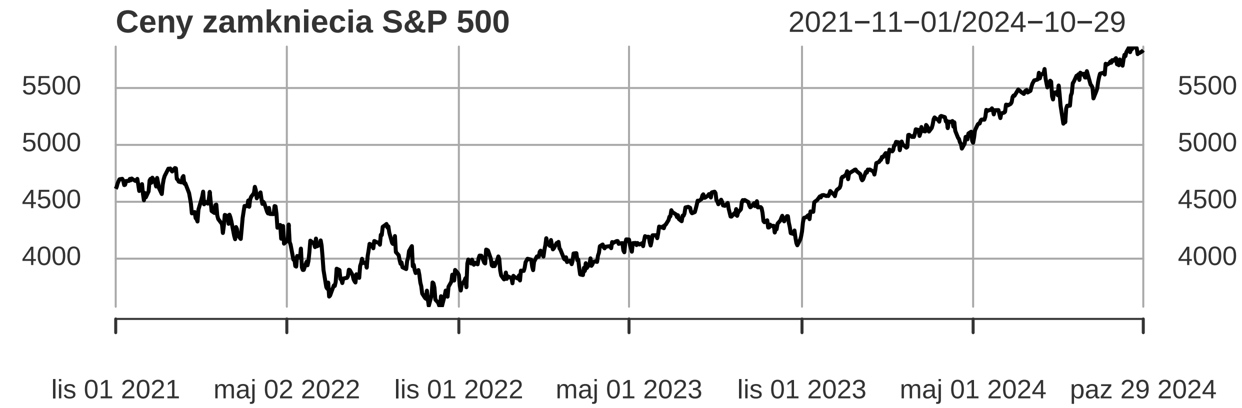
PKT 1

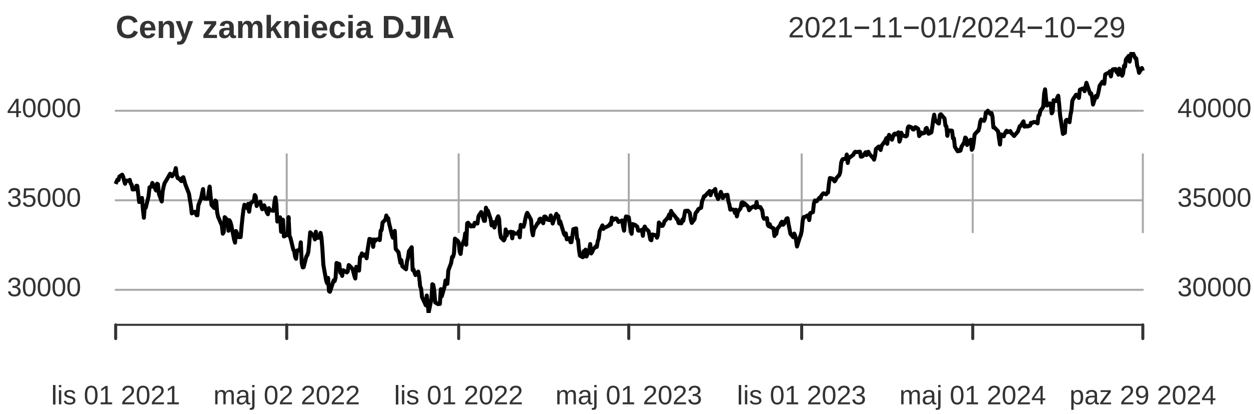
Dane, które dobieram do poprzednio wybranego indeksu S&P 500 to indeks giełdowy DJIA, czyli indeks notujący wartości z tych samych giełd co S&P 500 i który notuje wartości dla jednych z największych amerykańskich przedsiębiorstw. Firmy te w większości również wliczają sie do indeksu S&P 500, stąd też zakładam, że notowania DJIA będą bezpośrednio powiązane z S&P 500, zwłaszcza jeśli dla obu wezme ceny zamknięcia w tym samym okresie. Wpierw sprawdzam ogólne charakterystyki danych:

```
##      Index              GSPC.Close
##  Min.   :2021-11-01   Min.   :3577
##  1st Qu.:2022-08-02   1st Qu.:4090
##  Median :2023-05-02   Median :4415
##  Mean   :2023-05-01   Mean   :4531
##  3rd Qu.:2024-01-31   3rd Qu.:4925
##  Max.   :2024-10-29   Max.   :5865
```

```
##      Index              DJI.Close
##  Min.   :2021-11-01   Min.   :28726
##  1st Qu.:2022-08-02   1st Qu.:33224
##  Median :2023-05-02   Median :34441
##  Mean   :2023-05-01   Mean   :35340
##  3rd Qu.:2024-01-31   3rd Qu.:37905
##  Max.   :2024-10-29   Max.   :43276
```

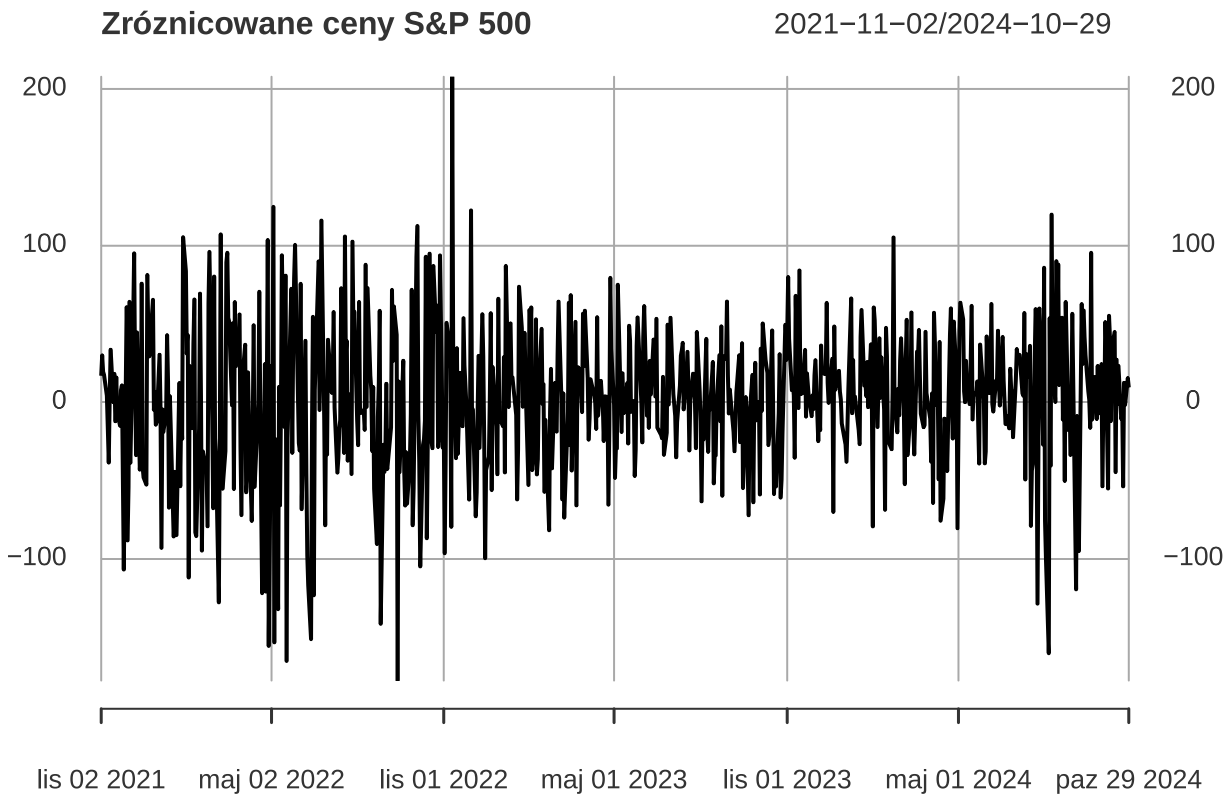
A następnie wyświetlam wykresy tych indeksów:



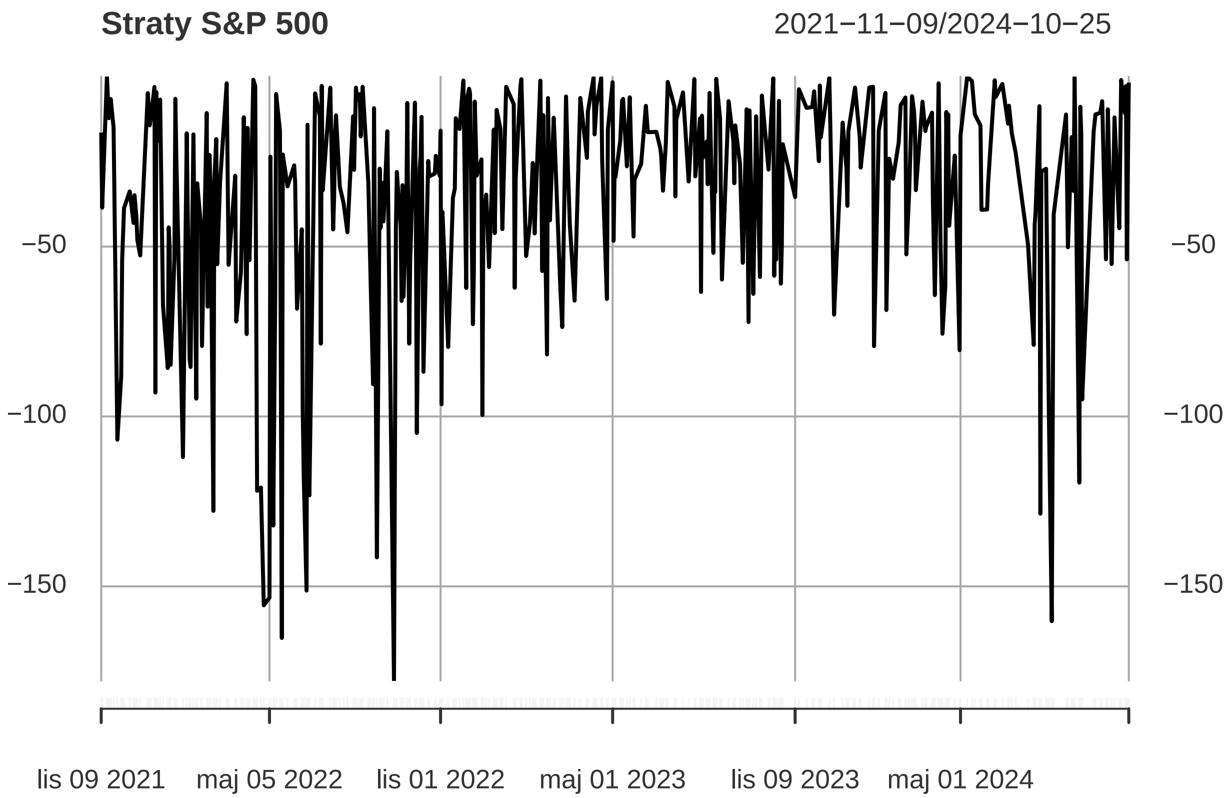


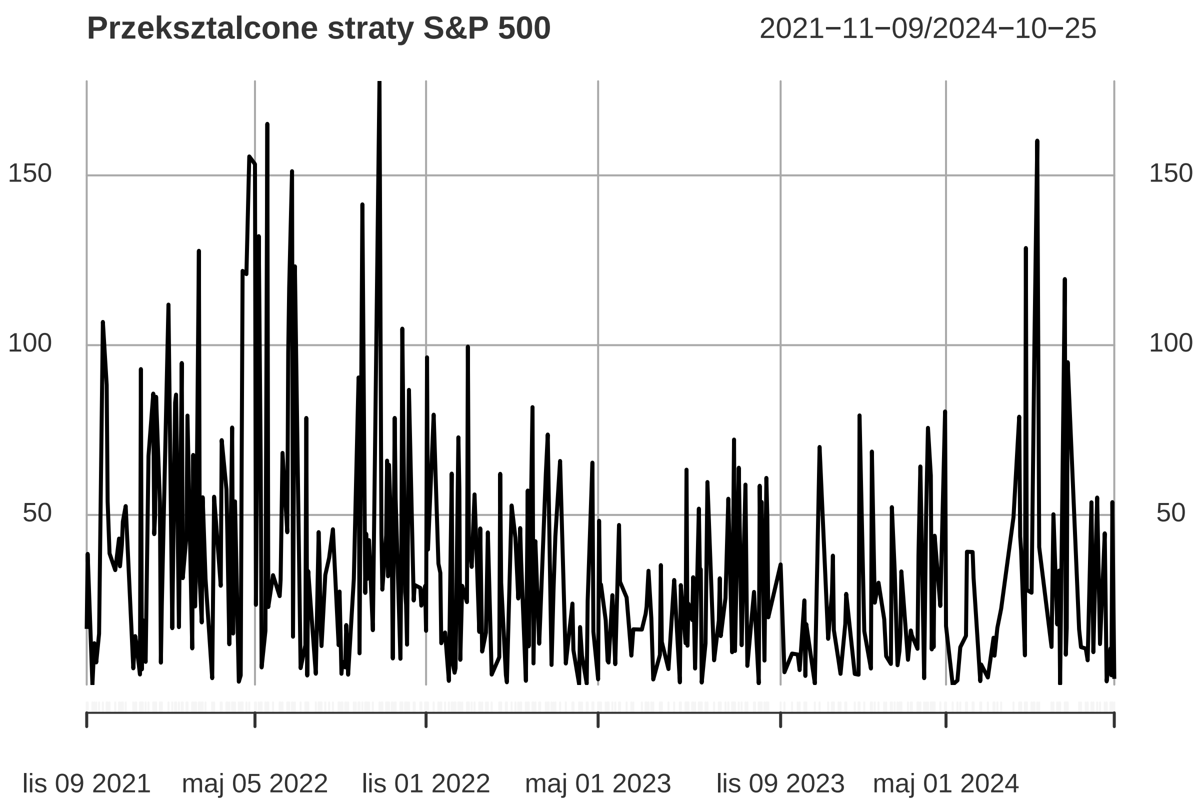
Oba wykresy mają bardzo podobny kształt do siebie, jedyne różnią się skalą wartości na osi Y. Jedyna drobna różnica, którą mogę zauważyc to lekko inne wartości w okresie tuż po 1 listopada 2022 (stąd też potwierdzam, że te dane nie są identyczne). Zatem założenie o ich powiązaniu, patrząc tylkona te wykresy, jest poprawne. ## PKT 2 i 3

Drugim krokiem będzie wyznaczenie szeregów czasowych strat dla obu tych indeksów. W tym celu wpierw różnicuje osobno oba szeregi, by otrzymać dzienne zwroty, następnie ponieważ chce mieć same wartości strat, czyli wartości poniżej zera to usuwam wszystkie zerowe i dodatnie wartości z szeregów. Na koniec by móc dopasowywać szereg do rozkładu lognormlanego, który przyjjmuje tylko wartości dodatnie, zmieniam znak wszystkich pozostałych wartości, i w ten sposób otrzymuje dodatni szereg czasowy strat.

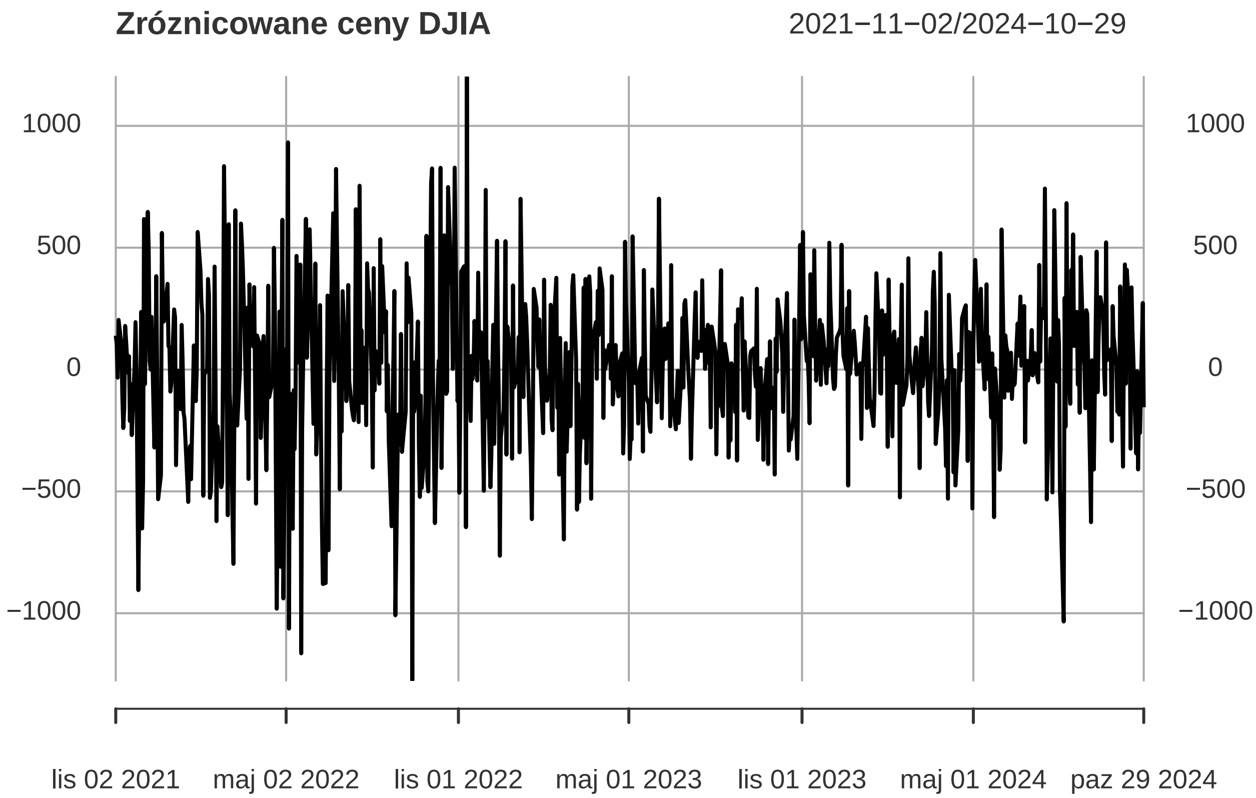


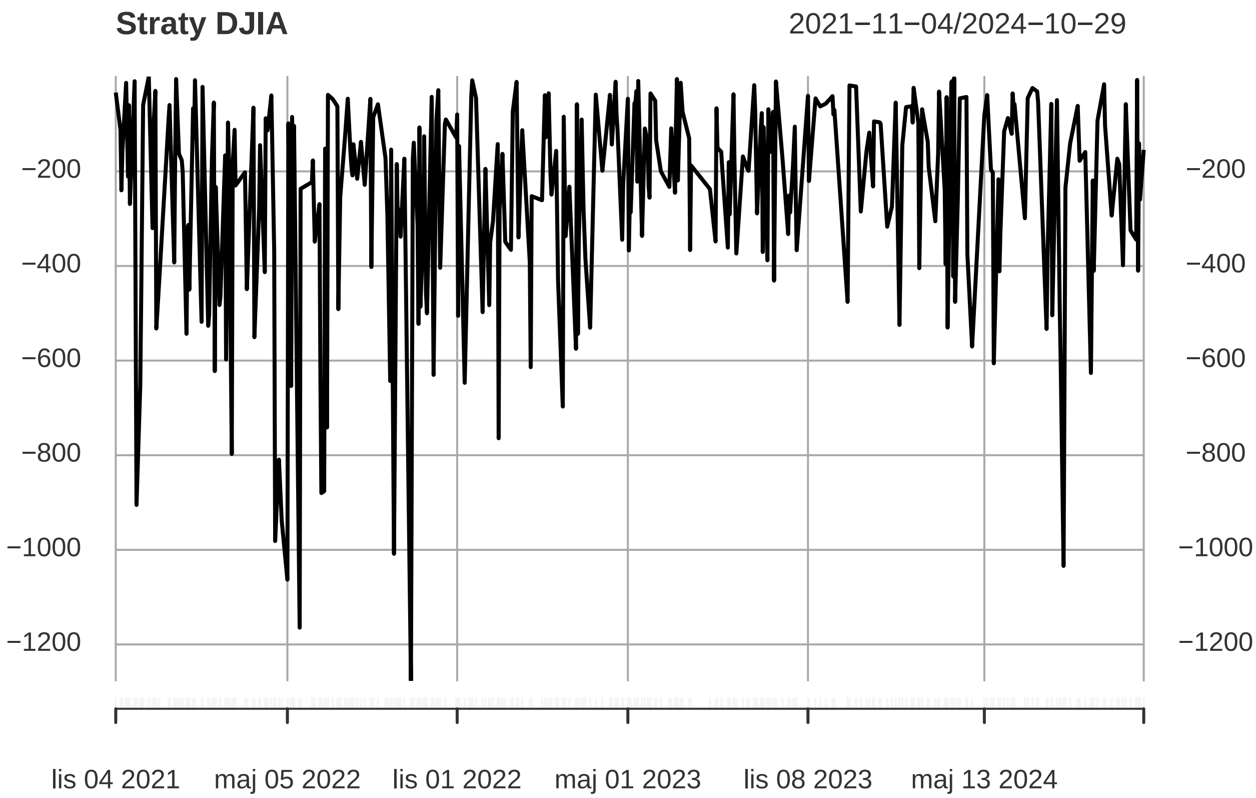
Zróznicowane ceny S&P 500
2021−11−02/2024−10−29
200
100
0
−100
200
100
0
−100
lis 02 2021
maj 02 2022
lis 01 2022
maj 01 2023
lis 01 2023
maj 01 2024
paz 29 2024


Straty S&P 500
2021−11−09/2024−10−25
−50
−100
−150
lis 09 2021
maj 05 2022
lis 01 2022
maj 01 2023
lis 09 2023
maj 01 2024

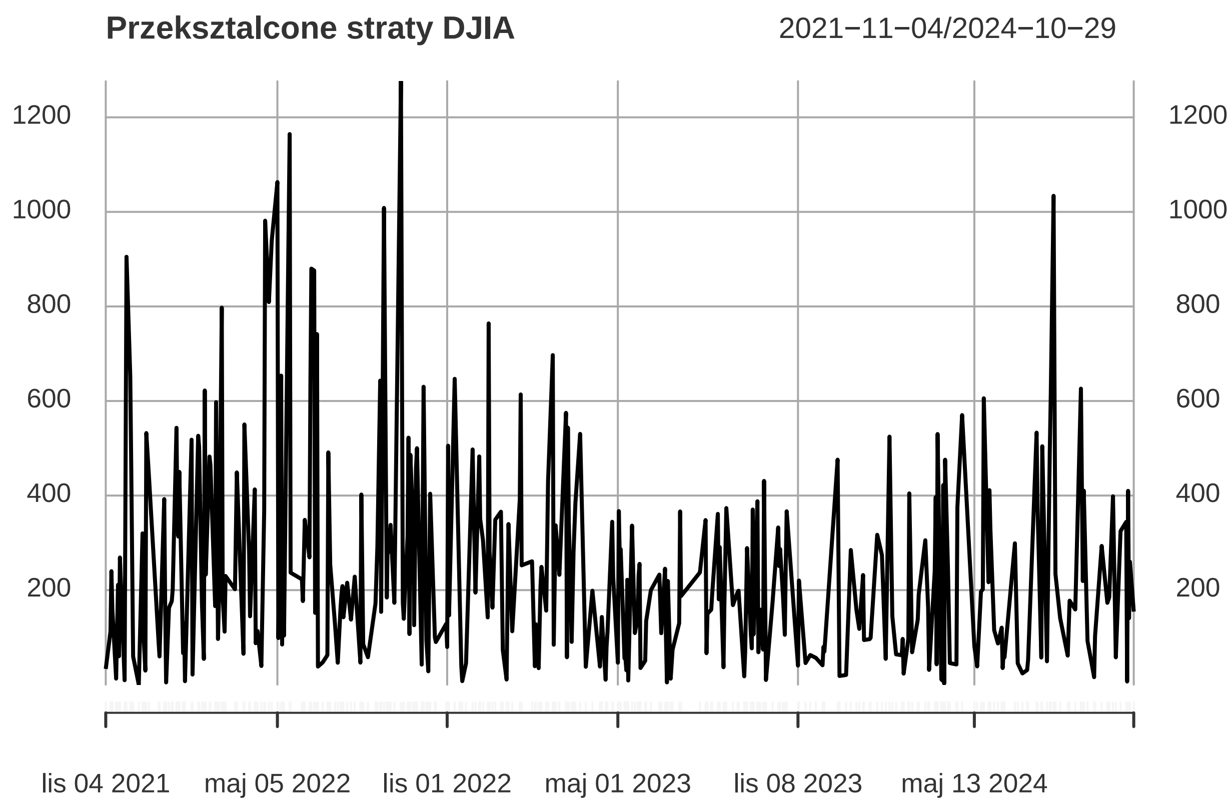


Przeksztalcone straty S&P 500
2021−11−09/2024−10−25
150
100
50
150
100
50
lis 09 2021
maj 05 2022
lis 01 2022
maj 01 2023
lis 09 2023
maj 01 2024

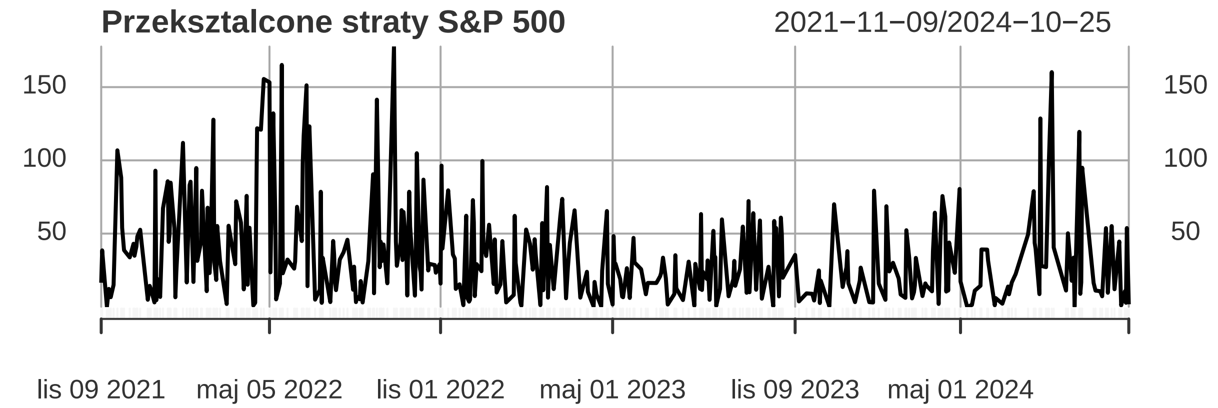

Zróznicowane ceny DJIA
2021−11−02/2024−10−29
1000
500
0
−500
−1000
1000
500
0
−500
−1000
lis 02 2021
maj 02 2022
lis 01 2022
maj 01 2023
lis 01 2023
maj 01 2024
paz 29 2024

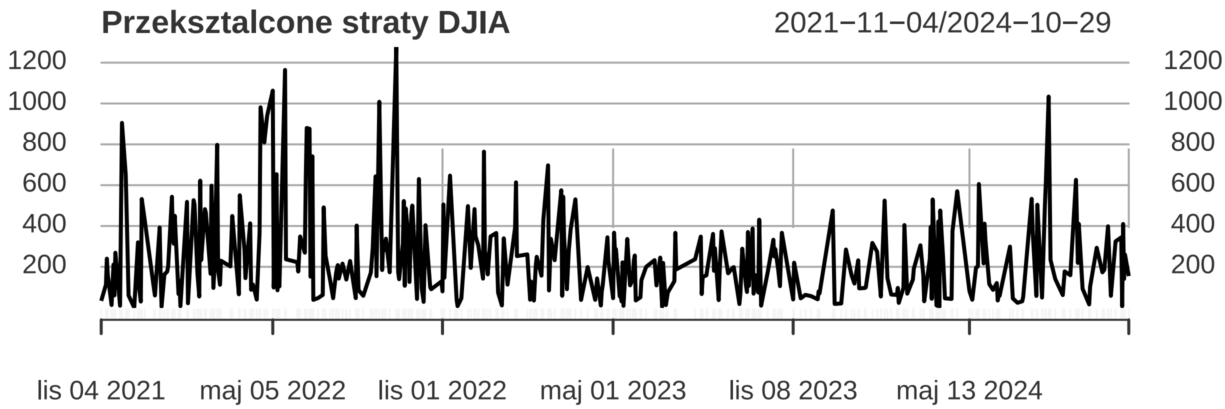


Straty DJIA
2021−11−04/2024−10−29
−200
−400
−600
−800
−1000
−1200
lis 04 2021
maj 05 2022
lis 01 2022
maj 01 2023
lis 08 2023
maj 13 2024

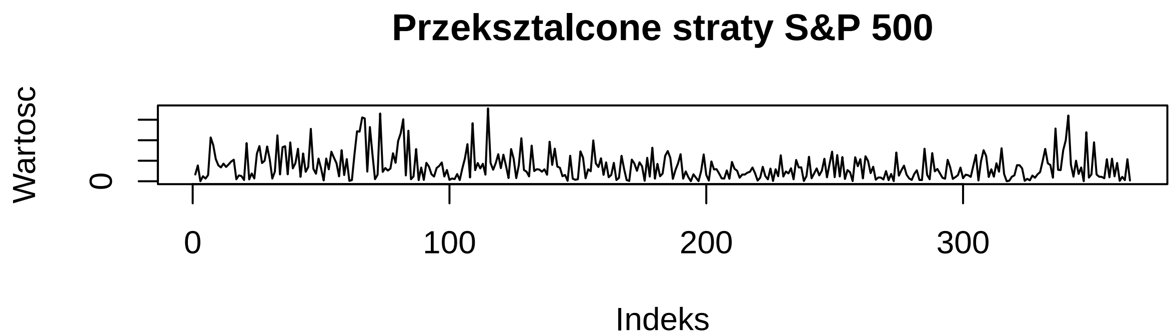


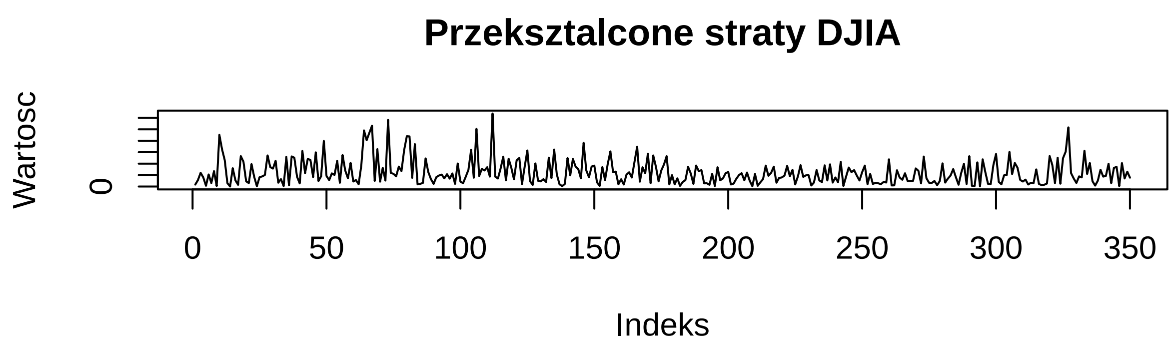
Mając już wyznaczone oba szeregi strat moge je ze sobą porównać:





Ponieważ same szeregi czasowe były podobne, to i również szeregi samych strat nie będą daleko od siebie odbiegały. Ważne jest jednak odnotowanie na tych wykresach, że oprócz różnej skali na osi Y, oś X również nie jest identyczna, co wynika z faktu że usuwając wszystkie dodatnie wartości ze zróżnicowanego szeregu zaburzyłem dzienną ciągłość danych, i sprawiłem że te szeregi nie mają równej długości. Najlepiej będzie to widać gdy przekształce je na zwykłe szeregi czasowe z numerycznymi indeksami:





Widać, że strat w przypadku S&P 500 jest więcej, natomiast ta różnica w ilości nie jest specjalnie duża, dlatego nadal mogę porównywać kształty tych wykresów. Tak samo jak dla oryginalnych szeregów czasowych są one bardzo podobne, czyli starty są ze sobą skorelowane.

Mając już wyznaczone szeregi strat wyświetlam ich podstawowe statystyki oraz wyznaczam ilość braków w danych, które interpretuje jako ilość danych usuniętych w poprzednich przeształcania oryginalnego zróżnicowania szeregu:

```
##      Index              GSPC.Close
##  Min.   :2021-11-09   Min.   :  0.03027
##  1st Qu.:2022-07-12   1st Qu.: 10.70996
##  Median :2023-03-01   Median : 27.10010
##  Mean   :2023-03-31   Mean   : 35.42667
##  3rd Qu.:2023-12-20   3rd Qu.: 50.02979
##  Max.   :2024-10-25   Max.   :177.72021
```

```
##             GSPC.Close
## średnia      35.42667
## odchylenie   33.18794
## wariancja  1101.43911
## moda         12.22998
## braki       388.00000
```

```
##      Index              DJI.Close
##  Min.   :2021-11-04   Min.   :   0.0586
##  1st Qu.:2022-07-01   1st Qu.:  79.7832
```

```
##  Median :2023-03-16   Median : 184.5742  
##  Mean   :2023-04-02   Mean   : 246.3740  
##  3rd Qu.:2024-01-01   3rd Qu.: 347.8525  
##  Max.   :2024-10-29   Max.   :1276.3691
```

```
##               DJI.Close
## średnia      246.373973
## odchylenie   220.736998
## wariancja  48724.822202
## moda           4.808594
## braki        403.000000
```

Widać, że dla S&P 500 jest 15 więcej wartości niż dla DJIA. Kolejnym krokiem będzie sprawdzenie korelacji między tymi szeregami. Nie można jednak tego zrobić dla nierównych szeregów czasowych. Dlatego, by rozwiązać te nierówność między długościami w najbardziej "sprawiedliwy sposób", wyselekcjonuje z obu szeregów tylko te wartości, które mają swój odpowednik w drugim szeregu (inaczej mówiąc wezmę pod uwagę wartości tylko z dni, gdzie oba indeksy zanotowały stratę). Wpierw jednak muszę się upewnić, że nie zredukuje to zbytnio mojej ilości danych:

```
## [1] "Długość nowych szeregów strat:"
```

```
## [1] 298
```

```
## [1] "Ilość straconych wartości dla dłuższego szeregu S&P 500:"
```

```
## [1] 67
```

Strata 67 wartości jest dosyć duża, ale nadal posiadam prawie 300 wartości do przeprowadzenia analizy, co nie wygląda niemożliwie do zrobienia. Sprawdzam więc korelacje między zmiennymi na podstawie współczynników:

```
## [1] "Współczynnik Pearsona"
```

```
##            DJI.Close
## GSPC.Close 0.3414489
```

```
## [1] "Współczynnik Spearmana"
```

```
##            DJI.Close
## GSPC.Close 0.1900543
```

```
## [1] "Współczynnik Kendalla"
```

```
##            DJI.Close
## GSPC.Close 0.1308701
```

Mimo tak podobych wykresów indeksów, jak i ich strat, każdy z współczynników wskazuje na nie aż tak znaczną korelacje między zmiennymi. Największa jest korelacja liniowa związana z współczynnikiem Pearsona, natomiast oczekiwałem tutaj wyniku znacznie bliżej jedynki. Tak samo dla zależności monotonicznej, czy według Spearmana czy Kendalla, wartość w moim odczuciu powinna wyjść większa. Sugeruje to, że albo ograniczenie się do samych strat, które jeszcze dodatkowo okroiłem, zbyt naruszyło zależności moich danych ze względu na zbyt małą próbke, albo że szeregi może i są podobne, ale wyizolowane straty i ich wartości nie są jednak tak samo zależne.

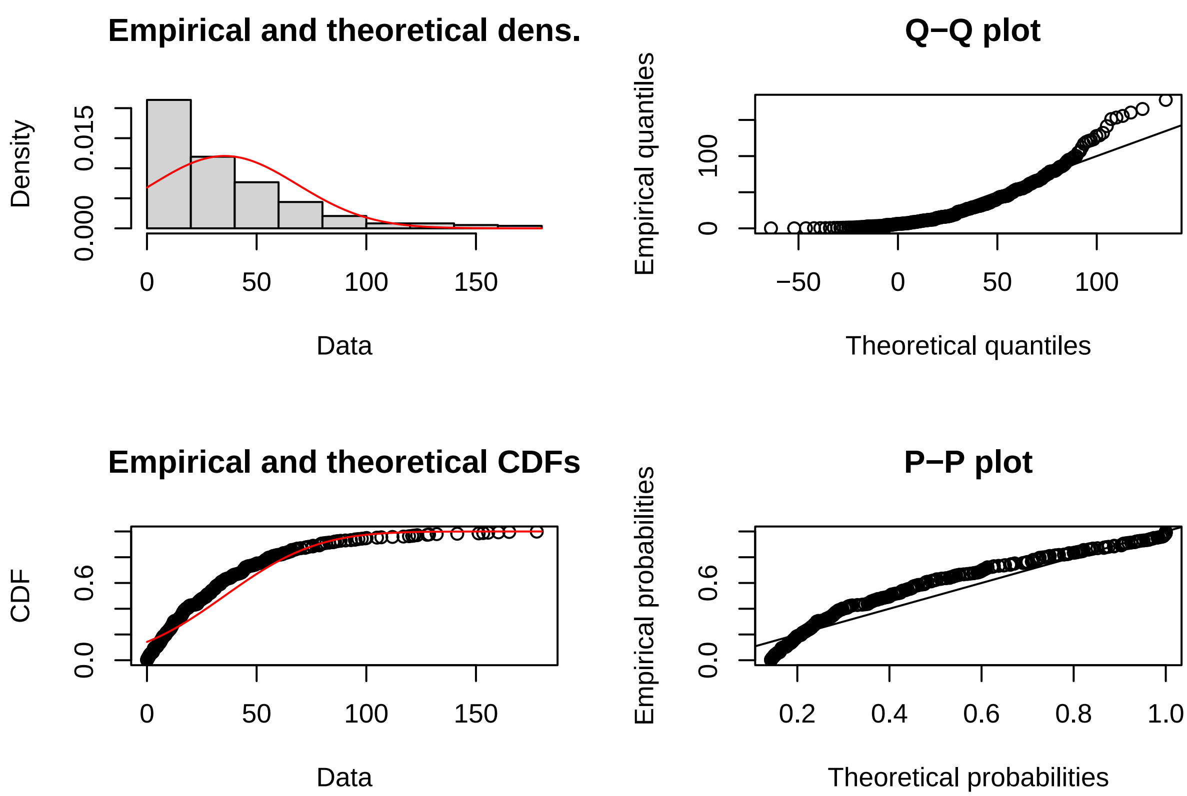
PKT 4

Kolejno dopasowuje do obu szeregów rozkład normalny i lognormalny i na podstawie wykresów diagnostycznych będe oceniał ich dopasowanie.

Wpierw szereg strat indeksu S&P 500. Parametry rozkładu normalnego:

```
##     mean       sd
## 35.42667 33.14244
```

Wykresy diagnostyczne rozkładu normalnego:

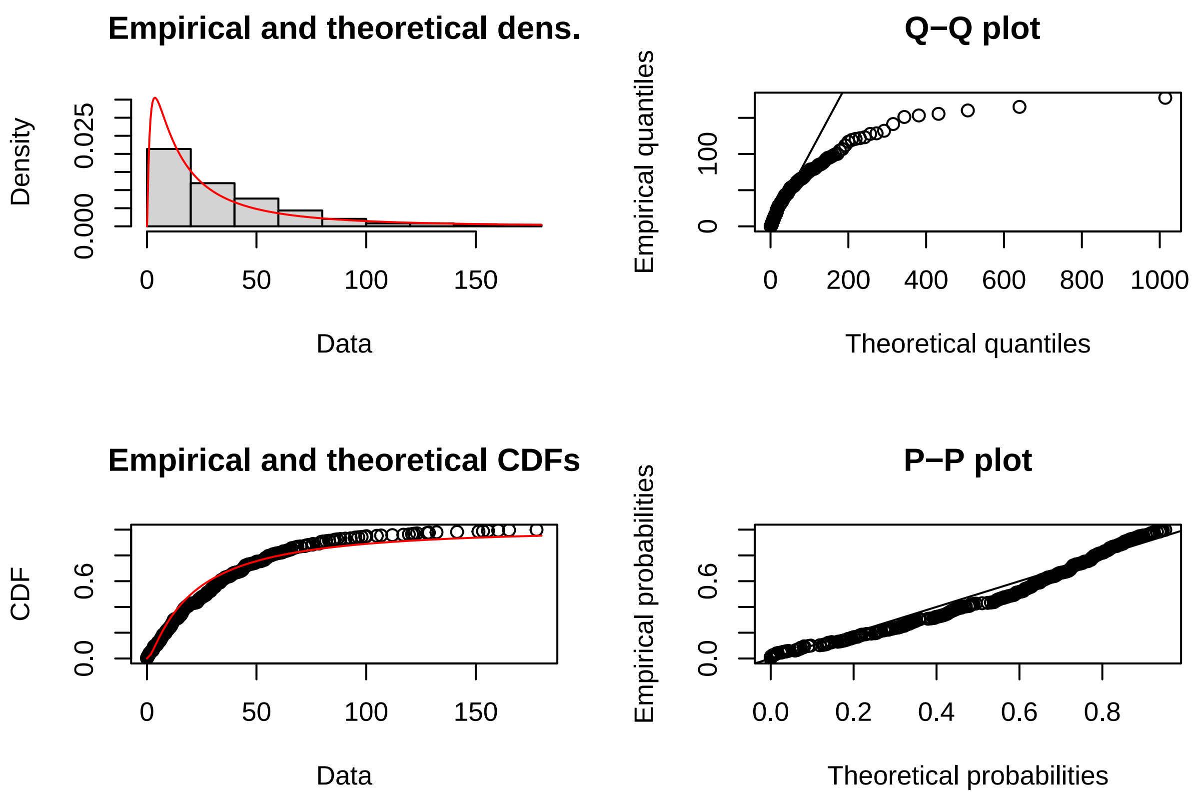


Sam histogram pokazuje, że dane nie przypominają rozkładu normalnego. Sam rozkład jest prawoskośny, z duża ilościa małych elementów koło 0 (czyli, że mam o wiele więcej drobnych strat niż załamań wartości). Pozostałe wykresy nie są również idealne, ale nie wyglądają tak źle, jak sugerowałby to histogram, co sugeruje, że rozkład może mieć jakiś związek z rozkładem normalnym (np. rozkład lognormalny).

Parametry rozkładu lognormalnego:

```
##  meanlog    sdlog
## 3.005755 1.307335
```

Wykresy diagnostyczne rozkładu lognormalnego:

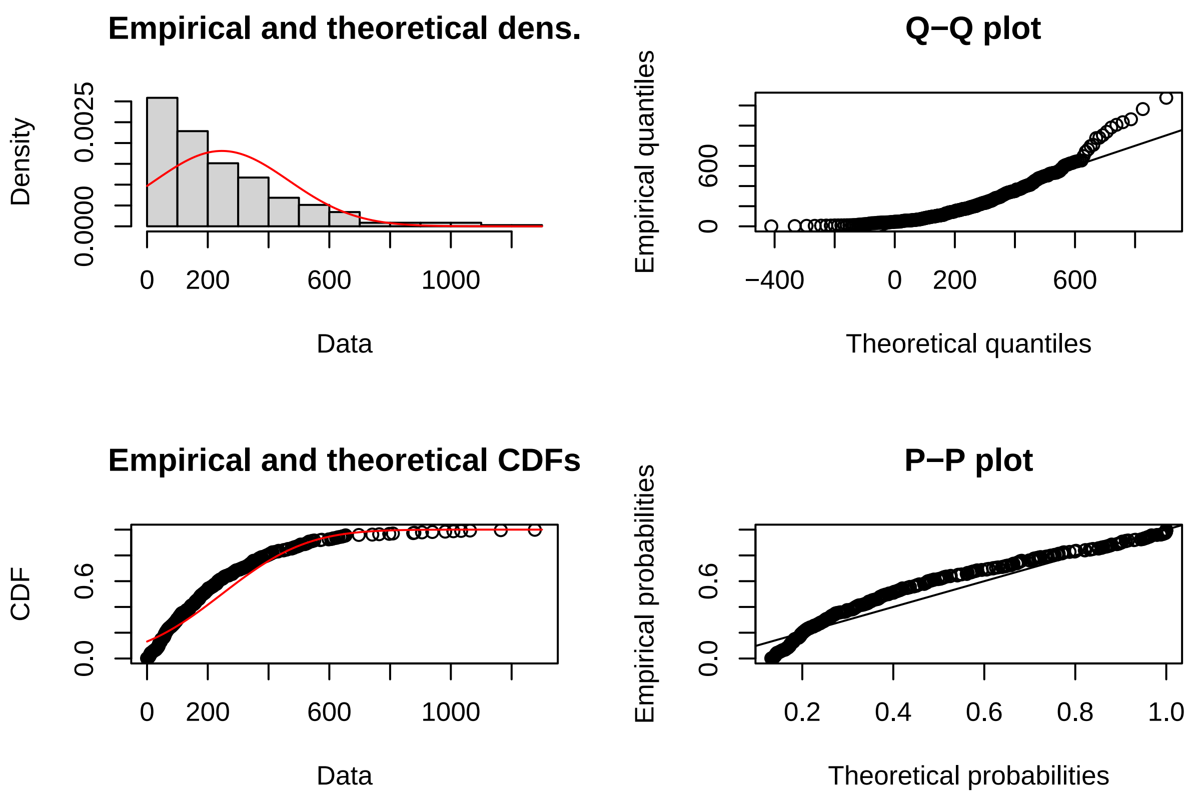


Histogram, jak i funkcja gęstości o wiele bardziej sugerują, że nasze dane pochodzą z rozkładu lognormalnego. Dwa z trzech wykresów diagnostycznych znacznie poprawiły się względem rozkładu normalnego. Najbardziej jednak dziwi wykres kwantyl-kwantyl, który wygląda jakby był jakimś błędem. Nie wiem jednak z czego mogłoby to wynikać (chyba że ze zbyt małej liczby danych).

Następnie szereg strat indeksu DJIA. Parametry rozkładu normalnego:

```
##     mean       sd
## 246.3740 220.4214
```

Wykresy diagnostyczne rozkładu normalnego:

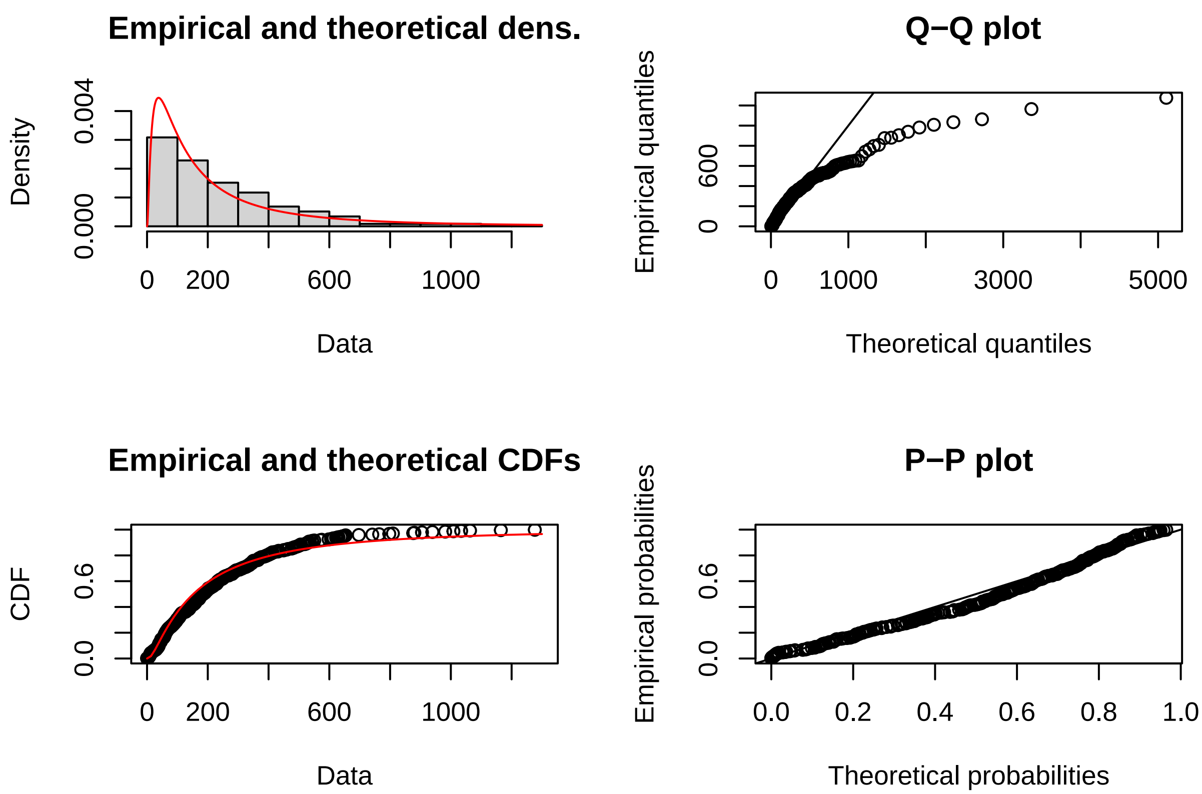


Ponownie można wyciągnać te same wnioski, prawoskośny rozkład, histogram nie przypomina rozkładu normalnego, ale same wykresy diagnostyczne nie odstraszają swoim wyglądem.

Parametry rozkładu lognormalnego:

```
##  meanlog    sdlog
## 5.023832 1.178210
```

Wykresy diagnostyczne rozkładu lognormalnego:



Ponownie wszystko poza wykresem kwantyl kwantyl sugeruje, że nasze dane pochodzą z rozkładu lognormalnego.

PKT 5

Kolejnym zadaniem będzie dopasowywanie kolejnych typów kopuł do połączonych szeregów strat. Wpierw połączmy je w jeden obiekt:

```
##       GSPC.Close  DJI.Close
## [1,] 0.354515050 0.08361204
## [2,] 0.648829431 0.34448161
## [3,] 0.003344482 0.60869565
## [4,] 0.280936455 0.43478261
## [5,] 0.163879599 0.05016722
## [6,] 0.311036789 0.55518395
```

```
##    GSPC.Close          DJI.Close
##  Min.   :0.003344   Min.   :0.003344
##  1st Qu.:0.251672   1st Qu.:0.251672
##  Median :0.500000   Median :0.500000
##  Mean   :0.500000   Mean   :0.500000
##  3rd Qu.:0.748328   3rd Qu.:0.748328
##  Max.   :0.996656   Max.   :0.996656
```

A następnie po kolei dopasowujmy je do kopuły typu:

1. Gumbela:

```
## Call: fitCopula(gumbel_copula, data = data_matrix)
## Fit based on "maximum pseudo-likelihood" and 298 2-dimensional observations.
## Gumbel copula, dim. d = 2
##       Estimate Std. Error
## alpha    1.188      0.057
## The maximized loglikelihood is 13.07
## Optimization converged
## Number of loglikelihood evaluations:
## function gradient
##        8        8
```

2. Franka:

```
## Call: fitCopula(frank_copula, data = data_matrix)
## Fit based on "maximum pseudo-likelihood" and 298 2-dimensional observations.
## Frank copula, dim. d = 2
##       Estimate Std. Error
## alpha    1.208      0.348
## The maximized loglikelihood is 5.656
## Optimization converged
## Number of loglikelihood evaluations:
## function gradient
##        4        4
```

3. Claytona:

```
## Call: fitCopula(clayton_copula, data = data_matrix)
## Fit based on "maximum pseudo-likelihood" and 298 2-dimensional observations.
## Clayton copula, dim. d = 2
##       Estimate Std. Error
## alpha   0.3012      0.066
## The maximized loglikelihood is -2.086
## Optimization converged
## Number of loglikelihood evaluations:
## function gradient
##        3        3
```

4. Normalna:

```
## Call: fitCopula(normal_copula, data = data_matrix)
## Fit based on "maximum pseudo-likelihood" and 298 2-dimensional observations.
## Normal copula, dim. d = 2
##       Estimate Std. Error
## rho.1   0.2118      0.055
## The maximized loglikelihood is 6.42
## Optimization converged
## Number of loglikelihood evaluations:
## function gradient
##        6        6
```

5. T-studenta:

```
## Warning in var.mpl(copula, u): the covariance matrix of the parameter estimates
## is computed as if 'df.fixed = TRUE' with df = 7.26978181989888

## Call: fitCopula(t_copula, data = data_matrix)
## Fit based on "maximum pseudo-likelihood" and 298 2-dimensional observations.
## t-copula, dim. d = 2
##       Estimate Std. Error
## rho.1   0.2099      0.066
## df      7.2698         NA
## The maximized loglikelihood is 7.869
## Optimization converged
## Number of loglikelihood evaluations:
## function gradient
##       13        6
```

Mając już wszystkie dopasowania sprawdzę, która dopasowana kopuła będzie najlepsza dla moich danych względem kolejnych kryteriów dopasowania modelu. Wpierw tworze osobne listy wartości dla danej kopuły dla każdego kryterium:

```
## [1] "Kryterium loglikelihood"

##  fit_gumbel   fit_frank fit_clayton  fit_normal       fit_t
##   13.067449    5.655655   -2.085503    6.420428    7.869496

## [1] "Kryterium AIC"

##  fit_gumbel   fit_frank fit_clayton  fit_normal       fit_t
##  -24.134898   -9.311311    6.171005  -10.840857  -11.738991

## [1] "Kryterium BIC"

##  fit_gumbel   fit_frank fit_clayton  fit_normal       fit_t
##  -20.437804   -5.614217    9.868099   -7.143763   -4.344804
```

I końcowo wyznaczam najlepiej dopasowaną kopułę względem:

1. Kryterium loglikelihood(im większa wartość, tym lepiej dopasowany model):

```
## [1] "fit_gumbel"

## [1] 13.06745
```

2. Kryterium AIC(im mniejsza wartość, tym lepiej dopasowany model):

```
## [1] "fit_gumbel"

## [1] -24.1349
```

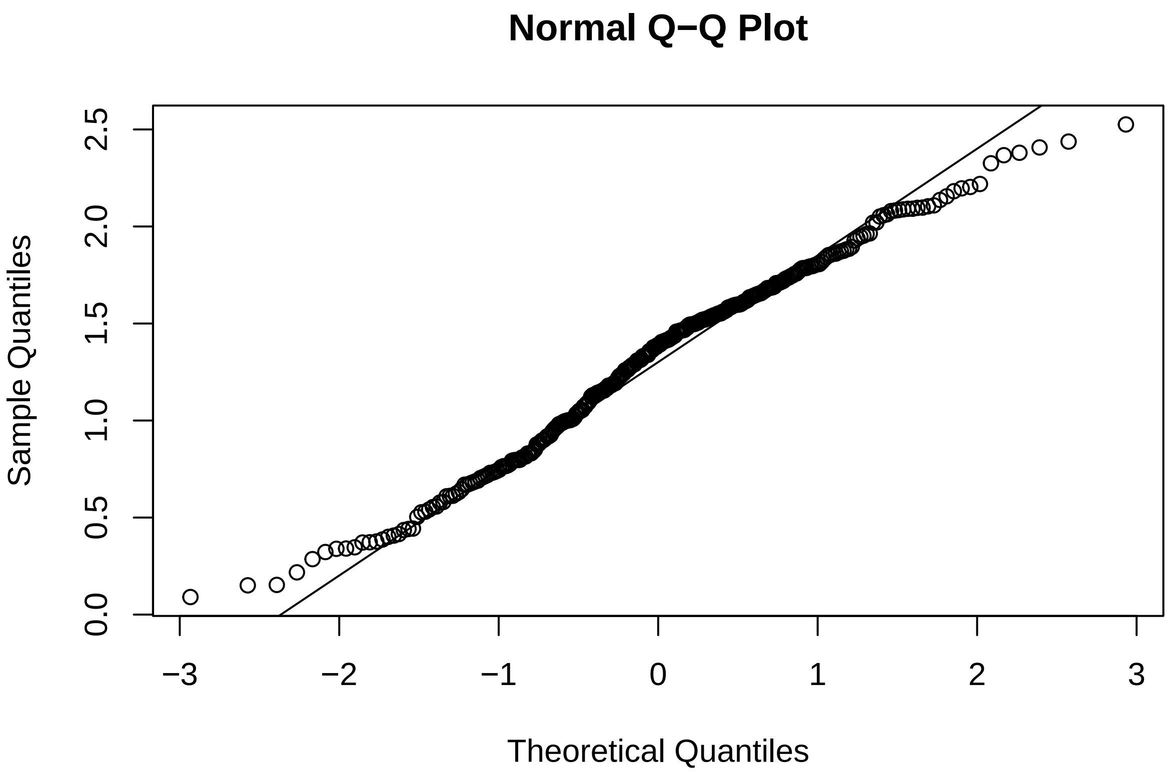
3. Kryterium BIC(im mniejsza wartość, tym lepiej dopasowany model):

```
## [1] "fit_gumbel"
```

```
## [1] -20.4378
```

Jak widać wszystkie 3 kryteria wskazały, że najlepiej dopasowaną kopułą do moich danych jest kopuła Gumbella.

Ostatnią rzeczą będzie wykonanie testu Mardia dla moich danych, który zbada wielowymiarową normalność między moimi danymi:




```
## Call: mardia(x = data_matrix)
##
## Mardia tests of multivariate skew and kurtosis
## Use describe(x) the to get univariate tests
## n.obs = 298   num.vars =  2
## b1p =  0.06   skew =  2.97  with probability  <=  0.56
##  small sample skew =  3.02  with probability <=  0.55
## b2p =  5.74   kurtosis =  -4.88  with probability <=  0.000001
```

Z wykresu kwantyl-kwantyl mogę odczytać, że kwantyle moich danych pasują do teoretycznych kwantyli dwuwymiarowego rozkładu normalnego, z pewnymi niedopasowaniami na ogonach. Z wartości testu mogę odczytać duże p-value (tutaj napisanie jako probability) dla skośności moich danych, co daje przyjęcie hipotezy zerowej, że moje dane są symetryczne jak wielowymiarowy rozkład normalny. Natomiast p-value dla kurtozy jest już poniżej poziomu istotności 0.05, co powoduje odrzucenie hipotezy zerowej, że moje dane mają kutroze podobną do wielowymiarowego rozkład normalnego, zatem końcowo test sugeruje że mimo symetryczności mojego rozkładu nie jest on wielowymiarowym rozkładem normalnym. Ten wniosek zgadza się z cała poprzednią analizą moich zmiennych.