Obliczanie VaR różnymi metodami

PW

5 kwietnia 2019

1. Wprowadzenie

Celem niniejszego projektu będzie wykorzystanie 3 metod pozwalających na wyznaczenie wartości narażonej na ryzyko (VAR) i próba odpowiedzi na pytanie, który z zaprezentowanych sposobów wyznaczania VAR (oraz ES, czyli warunkowej wartości oczekiwanej) jest najlepszy.

W ramach badania wykorzystam 3 metody :

- symulacji historycznej,
- bootstrap,
- symulacji historycznej z wagami.

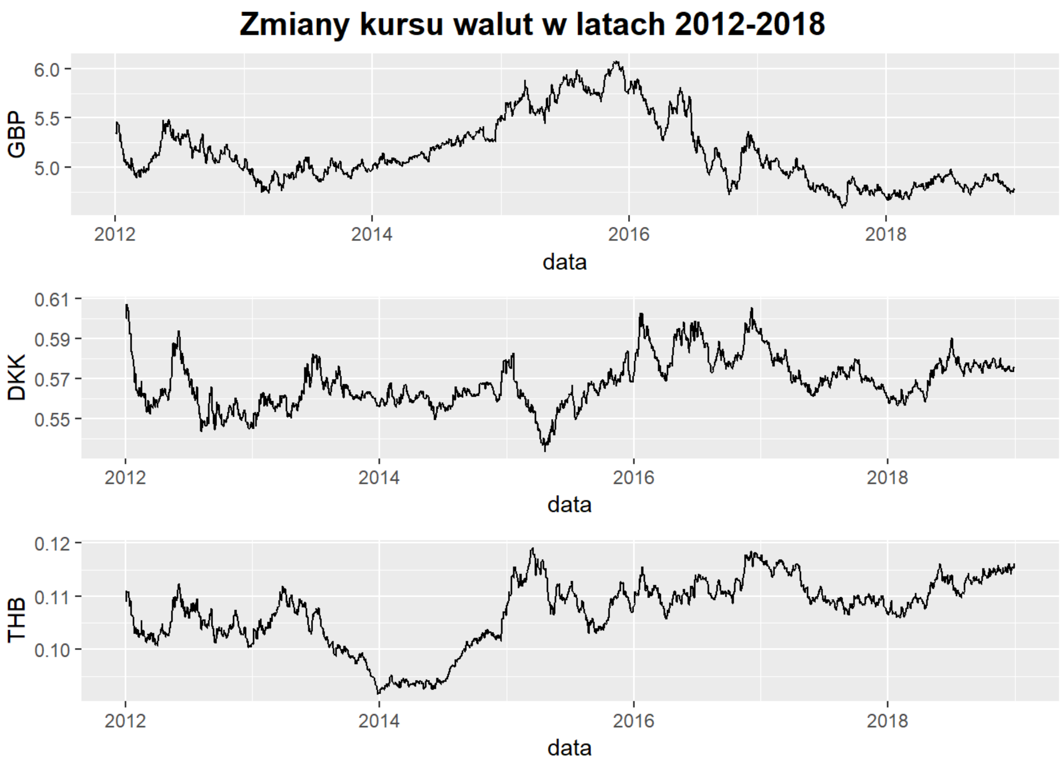
W każdej metodzie wyznaczania VaR będę wyznaczał go z 99% pewnością na 500dniowym oknie estymacji.

2. Opis danych

Do przeprowadzenia badania wykorzystam kursy walutowe z lat 2012 - 2018 dla 3 wybranych walut :

- **GBP**, czyli funta, waluty Wielkiej Brytanii,
- **DKK** czyli korony duńskiej obowiązującej na terenie Danii,
- oraz **THB**, czyli tzw. baht tajski lub inaczej bat, waluta obowiązująca w Tajlandii.

Kursy tych walut przedstawiały się następująco na przestrzeni rozważanego okresu:



Widzimy, że najsłabsza z walut jest rzecz jasna baht tajski, natomiast najmocniejszą, najbardziej cenną funt. Najstabilniejsza wydaje się jednak korona duńska, co potwierdzają wyliczone poniżej współczynniki zmienności (kolejno GBP, DKK i ostatni dla THB) dla naszych walut, gdzie zmienność waluty duńskiej kształtowała się na poziomie jedynie 2 %. Co ciekawe waluta brytyjska charakteryzuje się większą zmiennością niż bat, co jednak mniej dziwi, jeśli zwrócimy uwagę jak bardzo słabą, stosunkowo mało wartościową jest jednostką - 1 bat to równowartość około 10 - 12 groszy, podczas gdy funt wart był nawet 6 zł w 2016 roku, jednak brexit spowodował sukcesywny spadek jej wartości i dziś 1 funt kosztuje około 5 zł.

```
## [1] 6.74
```

```
## [1] 2.19
```

```
## [1] 5.89
```

Inne statystyki opisowe kształtują się następująco :

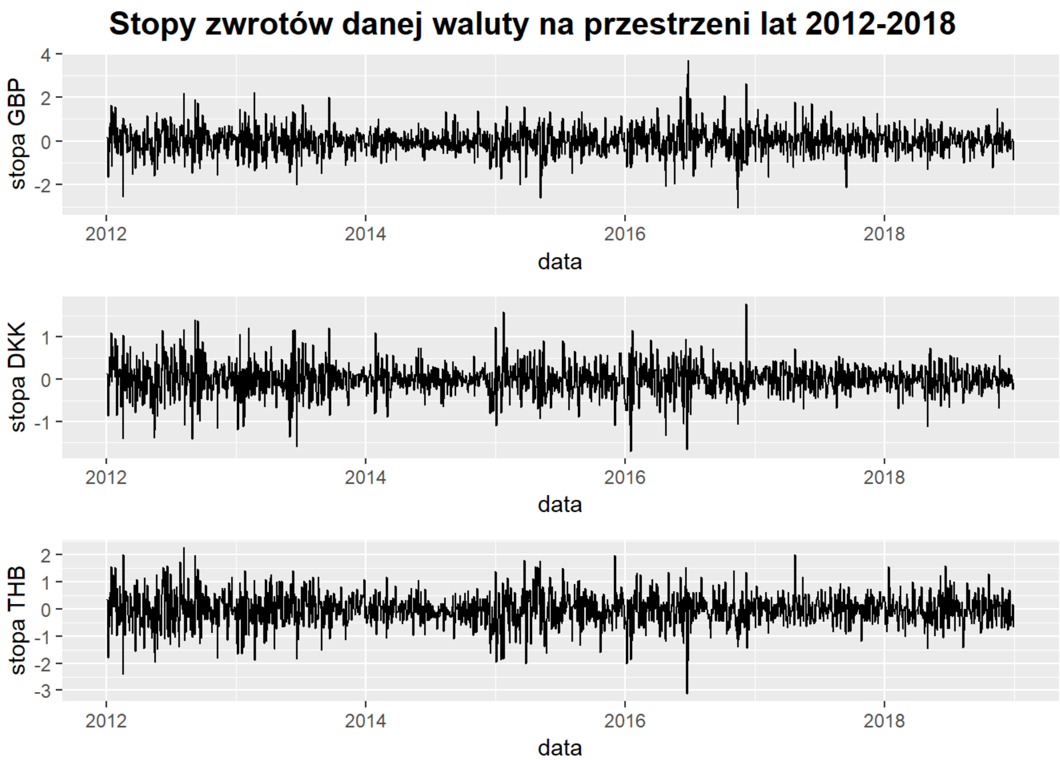
	mean	sd	median	min	max	range
X1GBP	5.1551463	0.3475943	5.06015	4.5852	6.0769	1.4917
X1DKK	0.5682473	0.0124388	0.56665	0.5338	0.6070	0.0732
X1THB	0.1071306	0.0063094	0.10850	0.0916	0.1192	0.0276

Widzimy, że zdecydowanie najwięcej zapłacić trzeba za funta, którego mediana kursów wynosi powyżej 5 zł, co jest wielokrotnością ceny za koronę duńską (56 groszy) czy bata (10 groszy). Na przestrzeni badanych 6 lat największa różnica między cenami funta wynosi niespełna 1,5 zł, wobec jedynie 7 groszy dla korony i prawie 3 groszy dla bata.

Do dalszej części projektu używał będę logarytmicznych stóp zwrotu danej waluty. Są odporniejsza na wahania, co stanowi ich przewagę nad stopami prostymi. Logarytmiczne stopy zwrotu pisują procentową zmianę ceny danej waluty w stosunku do dnia poprzedniego. Obliczam je za pomocą poniższej funkcji:

```r
# data - wektor cen walut w czasie
stopy_zw <- function(data){
  res <- NA
  for (i in 1:length(data)-1){
  res[i] <- 100*log(data[i+1]/data[i])
  #procentowe stopy zwrotu !
  }
  return(res)
}
```

Jak zaznaczyłem w komentarzu, stopy zwrotu podane będą w procentach. **Ponadto, wartości stóp mnożę po wyliczeniu przez -1, tak aby strat mieć podane w wartościach większych od 0. Pociąga to za sobą również konieczność szukania kwantyli 0.99, a nie 0.01 itd.** Jest to spowodowane faktem, że (przynajmniej mi) dzięki temu łatwiej patrzy się na wykresy i intepretuje wyniki. Tak więc jeśli w dalszej części projektu odnoszę się do stóp zwrotów lub strat, to mam na myśli logarytmiczne stopy zwrotu każdej waluty, ale z przeciwnym znakiem.



Poniżej statystyki opisowe:

	mean	sd	median	min	max	range
stopy_GBP	0.0062562	0.5873922	0	-3.048816	3.671648	6.720464

	mean	sd	median	min	max	range
stopy_DKK	0.0023820	0.3629795	0	-1.695667	1.783232	3.478899
stopy_THB	-0.0033198	0.5729170	0	-3.140680	2.283639	5.424319

Największe różnice między stopami miały miejsce jeśli chodzi o funta - rekordowy spadek to 3,67%, przy największym wzroście wynoszącym 3%. Potwierdza się, że najstabilniejszą walutą jest korona duńska, której rozstęp wynosi jedynie 3,48% przy powyżej 5,4% rozstępie waluty tajskiej oraz aż 6,72% w przypadku funta brytyjskiego. Największy zysk na wartości w stosunku do dnia poprzedniego to 3,14% w przypadku bahta tajskiego, a największy spadek to 3,67% z dnia 27 czerwca 2016 dla waluty brytyjskiej, co było spowodowane wynikami przeprowadzonego 4 dni wcześniej referendum, w którym obywatele Wielkiej Brytanii opowiedzieli się na opuszczeniem szeregów Unii Europejskiej przez ich kraj. Niepewność związana z dalszymi losami Wielkiej Brytanii widoczna była natychmiastowo na rynku walut, gdzie funt zaczął tracić na wartości. Zawirowania polityczne takiej jak właśnie brexit oraz fakt, że spośród badanych państw Wielka Brytania jest ‘najgrubszą rybą’ są powodem dużej zmienności w zakresie kursów tej waluty.

3. Opis metod i przedstawienie funkcji stworzonych do ich zastosowania

3.1 Metoda symulacji historycznej

Metoda symulacji historycznej bezpośrednio wykorzystuje dane historyczne do wyznaczenia VaR, poprzez wyznaczenie rozkładu stóp zwrotu i znalezienie odpowiedniego kwantyla. Zakłada ona, że zmiany, które wystąpią w przyszłości będą jednym z scenariuszy, które miały już miejsce w przeszłości. Zakłada również, że prawdopodobieństwo wystąpienia każdego ze scenariuszy jest identyczne.

Do wyznaczania VaR za jej pomocą wykorzystam poniższą funkcję, gdzie data to wektor zawierający stopy zwrotu, q jest kwantylem za pomocą którego wyznaczamy VAR, a d to argument określający szerokość okna estymacji. Do stworzonych wektorów VAR i ES będę przypisywał wartości wyliczone dla kolejnych dni za pomocą pętli, która odpowiada również za przesuwanie okna estymacji dla kolejnych dni. VAR obliczany będzie jako kwantyl z naszego okna estymacji, a Es jako średnia wartość z obserwacji większych lub równych naszemu kwantylowi.

Funkcja:

```
#q -> kwantyl, d -> szerokosc okna
symulacja_hist <- function(data, q = .99, d = 500){
  var <- NA
  ES <- NA
  for (i in 1:(length(data)-d)){
    okno <- data[i:(i+500-1)]
    var[i] <- quantile(okno, q)
    ES[i] <- mean(okno[okno >= quantile(okno, q)])
  }
  return(list(var = var,ES = ES))
}
```

Funkcja zwraca listę 2 elementową, której 1 element to wektor VAR, a drugi to rzecz jasna ES. Wywołanie funkcji oraz struktura zwracanego przez funkcję obiektu przedstawia się tak jak poniżej. Analogicznie będzie wyglądało to w przypadku pozostałych metod, dlatego nie będzie już później pokazywane.

```
var_sh_GBP <- symulacja_hist(data$stopy_GBP)
var_sh_DKK <- symulacja_hist(data$stopy_DKK)
var_sh_THB <- symulacja_hist(data$stopy_THB)

#struktura zwracanego przez funkcję obiektu.
str(var_sh_DKK)
```

```
## List of 2
##  $ var: num [1:1263] 1.18 1.18 1.18 1.18 1.18 ...
##  $ ES : num [1:1263] 1.28 1.28 1.28 1.28 1.28 ...
```

3.2 Metoda bootstrapowa

Bootstrap to metoda polegająca na wykorzystaniu losowania ze zwracaniem, by stworzyć wiele alternatywnych historii, na podstawie przeszłości. Szczególnie sensowne wydaje się jej zastosowanie, gdy dysponujemy małą ilością danych. VAR wyznaczany jest jako średnia z uzyskanych wartości VAR dla wylosowanych scenariuszy. Jej dodatkowym plusem jest fakt, że jej specyfika pozwala na określenie przedziałów ufności dla VAR. W przypadku tej metody, sama procedura liczenia VaR dla poszczególnych okien estymacji jest taka sama jak w przypadku metody historycznej. Jednak dzięki wielokrotnemu liczeniu VAR dla danego dnia (przy losowaniu różnych historii dla okien estymacji) metoda ta daje nadzieję na nieco dokładniejsze oszacowanie liczonych przez nas wartości.

Funkcja:

```
#n - liczba symulacji, n - liczba obserwacji w losowanej probce
bootstrap_var <- function(data, q = .99, d = 500, x = 600, n = 1000){
  var <- NULL
  ES <- NULL

  for (i in 1:(length(data)-d)){
    okno <- data[i:(i+500-1)]
    var_b <- NA
    ES_b <- NA

    for (i in 1:n){
    #losowanie indeksów [ze zwracaniem]
    set.seed(i)
    index <- sample(x = c(1:d), size = x, replace = TRUE)
    boot_sample <- okno[index]
    var_b[i] <- quantile(boot_sample, q)
    ES_b[i] <- mean(boot_sample[boot_sample >= var_b[i]])
    }

    var <- c(var, mean(var_b))
    ES <- c(ES, mean(ES_b))

  }
  return(list(var = var,ES = ES))
}
```

Za pomocą 2 pętli tworzę procedurę bootstrapu: 2ga pętla odpowiada za tworzenie 'n' zbiorów historycznych, dla każdego dnia, w którym wyznaczamy VAR. Dla każdego z wylosowanych zbiorów wyznaczamy VAR i ES tak jak w metodzie historycznej. Po wyznaczeniu 'n' (czyli 1000) wartości wyliczamy średnią z otrzymanych wyników i rozpoczynamy procedurą dla kolejnego okna estymacji. Z racji tego, że losujemy ze zwracaniem, to liczba obserwacji w wylosowanych zbiorach może być większa od liczby dni w naszym podstawowym oknie estymacji, dlatego domyślnie ustawiłem ją jako 600 - argument x. Argument n odpowiada za liczbę alternatywnych zbiorów losowanych przy wyznaczaniu VaR na dany dzień.

3.3 Metoda symulacji historycznej z wagami

Metoda symulacji historycznej z wagami to sposób wyznaczania VaR będący swojego rodzaju rozszerzeniem, udoskonaleniem metody symulacji historycznej. Uwzględnia fakt, że obserwacje, które miały miejsce niedawno, w lepszym stopniu opisują zmienność obecną zmienność oraz sytuacje rynkową. Tzw. 'wagi' są wyznaczane jako kolejne wyrazy ciągu geometrycznego o `q` bliskim 1. W moim badaniu `q` będzie równe 0.995, dzięki czemu im starsza obserwacja tym mniejsza będzie jej waga, lecz przy tym te najstarsze obserwacje nie będą zupełnie nieważne. Wzór na wagę 'i-tej' obserwacji przedstawia się następująco:

$$waga(i) = \frac{q^{n-i} * (1-q)}{1-q^n}$$

gdzie jako 'n' oznaczono szerokość okna estymacji.

Funkcja:

```
symulacja_hist_z_wagami <- function(data, q = .99, d = 500, g = .995){

  wagi <- sapply(c(1:d), function(x){(g^(d-x)*(1-g))/(1-g^d)})
  ES <- NA
  var <- NA

  for (i in (1:(length(data)-d))){
    okno <- data[i:(i+d-1)]
    rozklad <- DiscreteDistribution(supp = okno, prob = wagi)
    var[i] <- rozklad@q(q)
    index <- which(okno > var[i])
    ES[i] <- sum(okno[index]*(wagi[index])/(1-q)) +
     okno[which(okno == var[i])]*(1 - sum(wagi[index])/(1-q))
  }
  return(list(var = var,ES = ES))
}
```

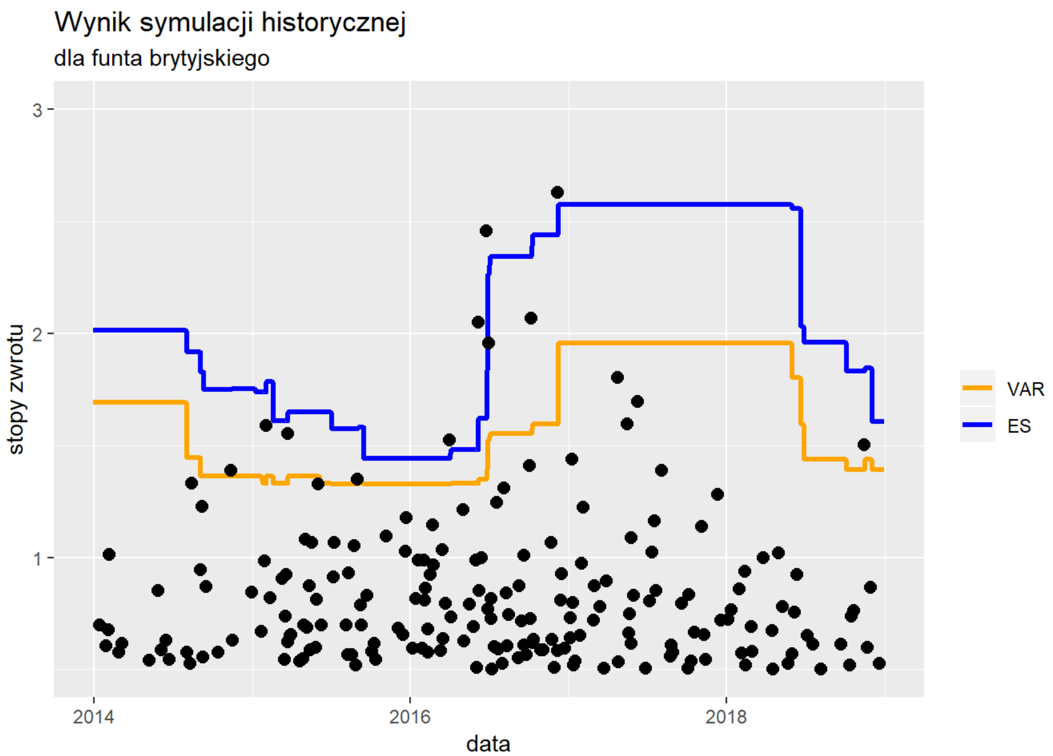
Najpierw utworzony zostaje wektor wag, zgodnie z wzorem podanym powyżej funkcji. Następnie za pomocą funkcji

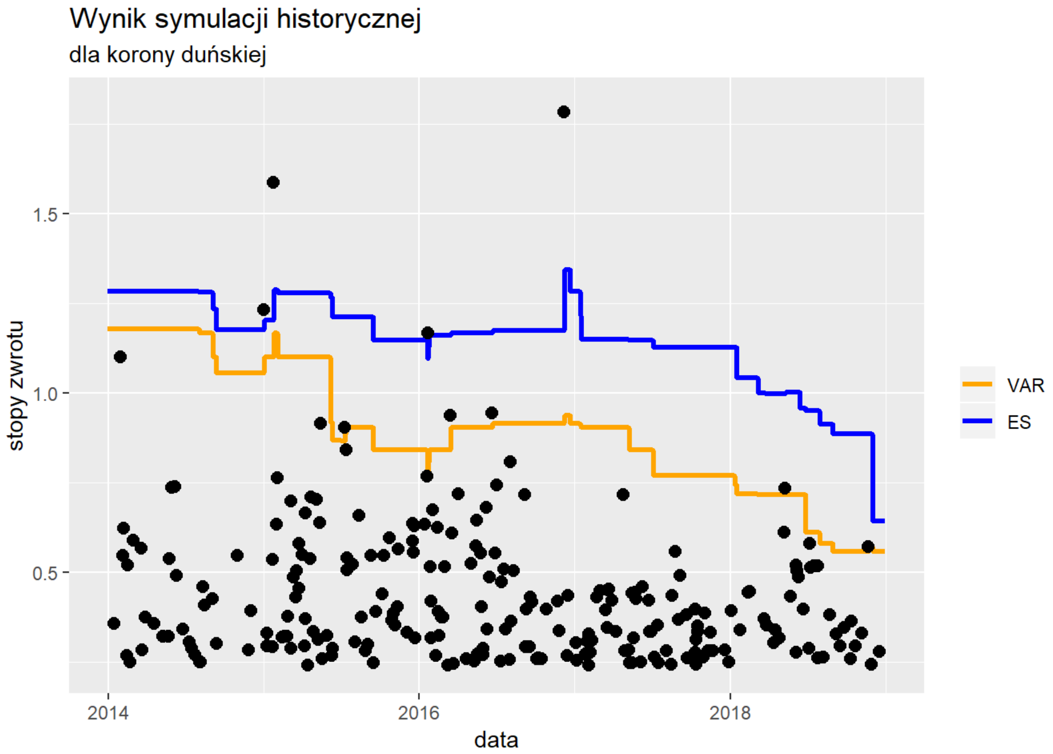
`DiscreteDistibution` tworzony jest rozkład z odpowiednimi wagami dla poszczególnych obserwacji. Następnie wyznaczany jest VAR (analogicznie jak w poprzednich metodach) oraz ES - który wyznaczany jest jako suma iloczynów obserwacji przekraczających wartość VAR w obecnym oknie estymacji i odpowiadającym im warunkowych prawdopodobieństw ich wystąpienia w wypadku gdy VAR jest przekroczony.

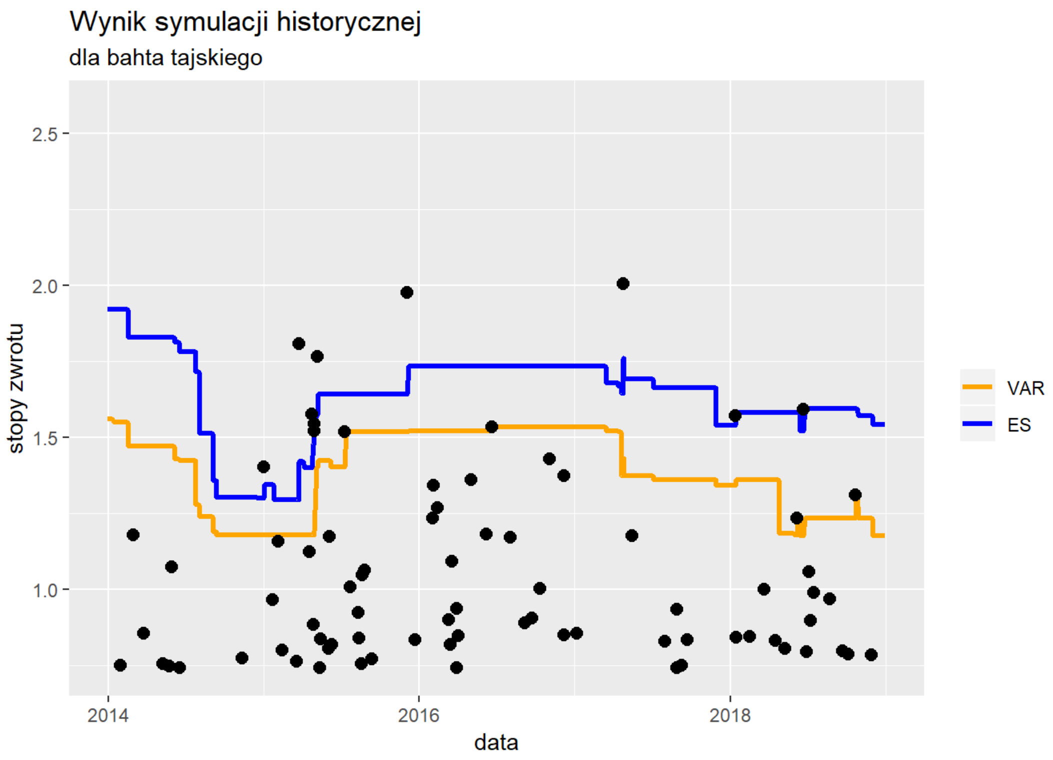
4. Przedstawienie wyników

Wyniki poszczególnych algorytmów przedstawione zostaną na wykresach. Dla każdej metody przedstawione zostaną wykresy, na których pokazane będą wyznaczone wartości VAR i ES, na tle strat (czyli stóp zwrotu pomnożonych przez -1).

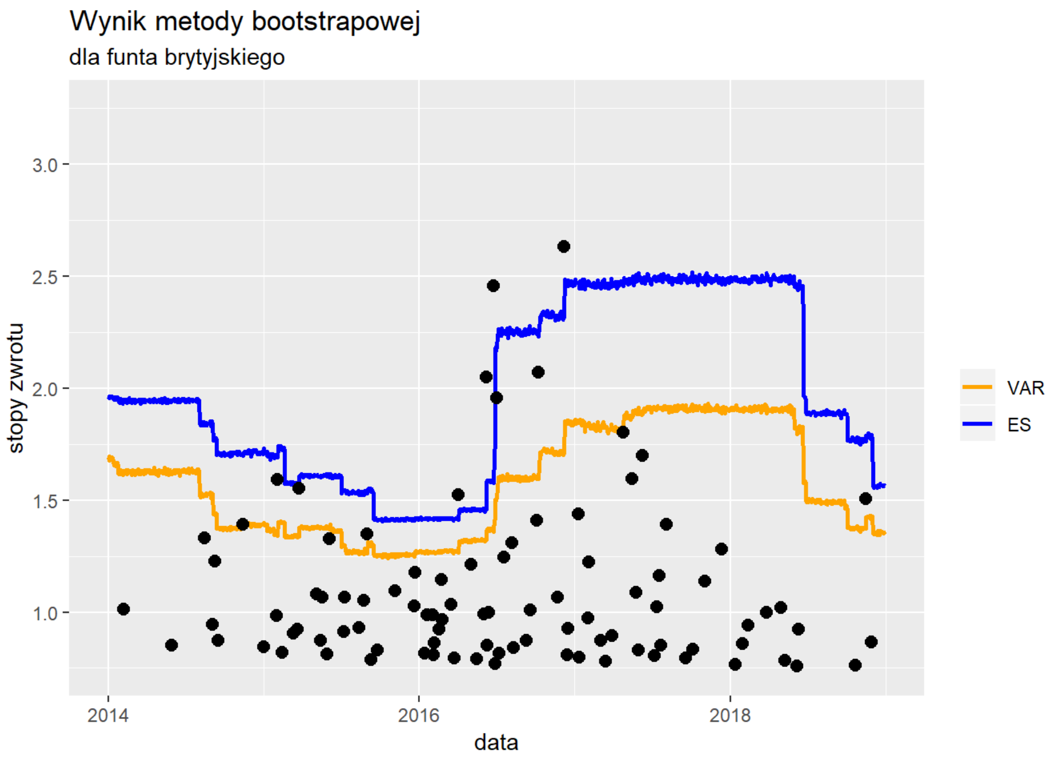
4.1 Dla metody historycznej

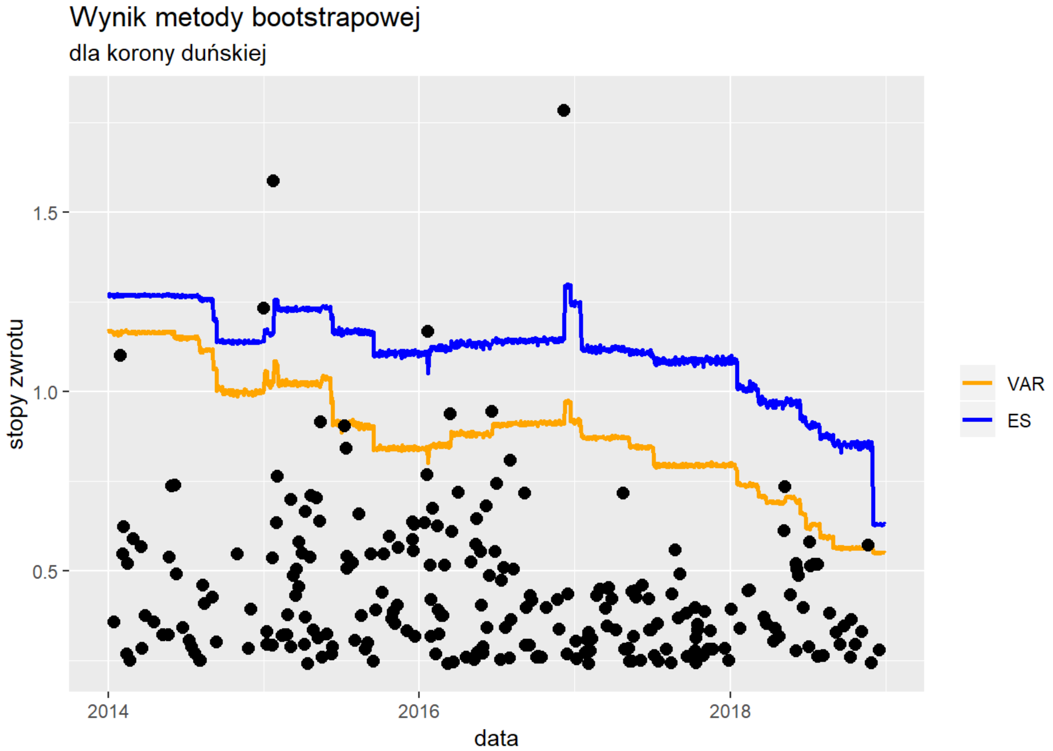


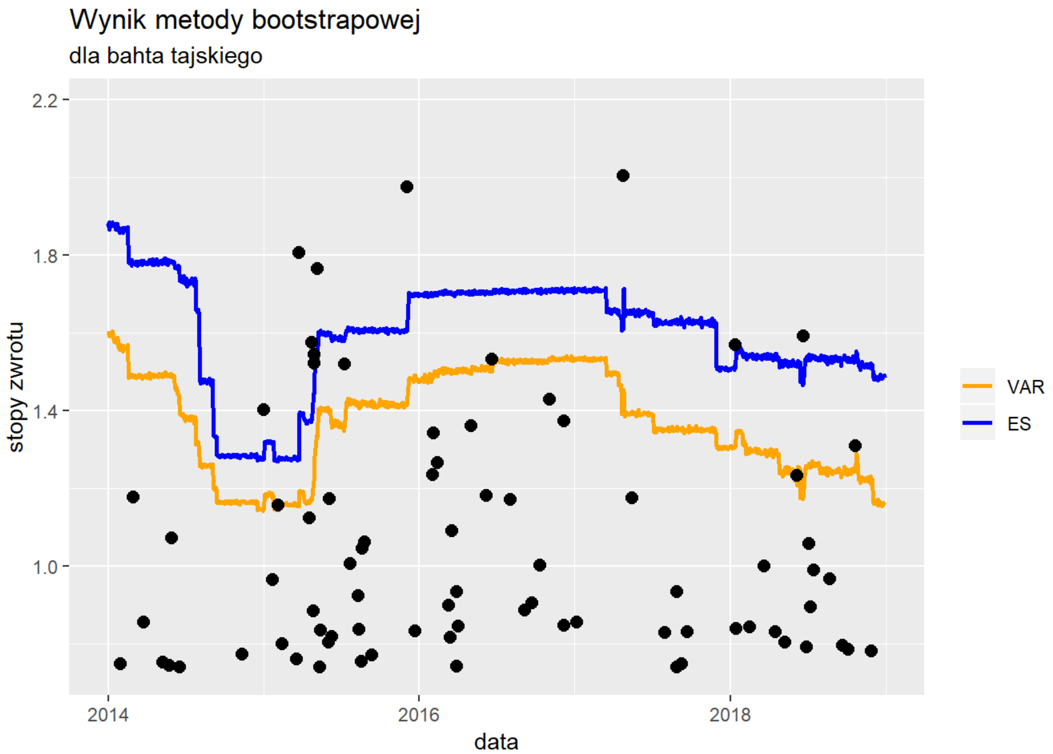




4.2 Dla metody bootstrapu



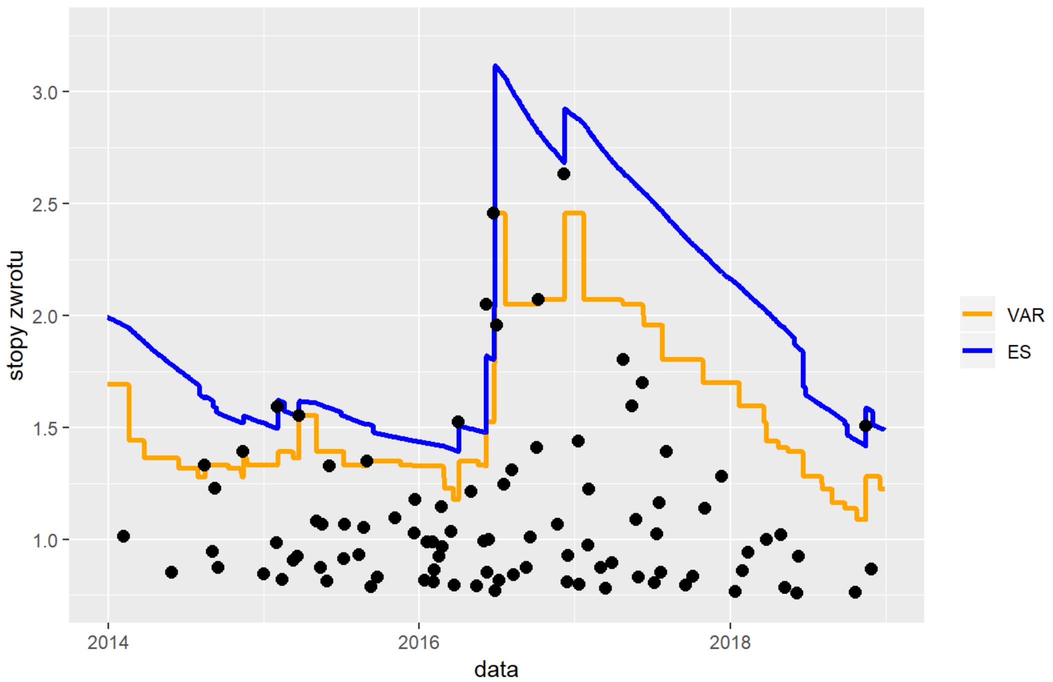




4.3 Dla metody symulacji historycznej z wagami

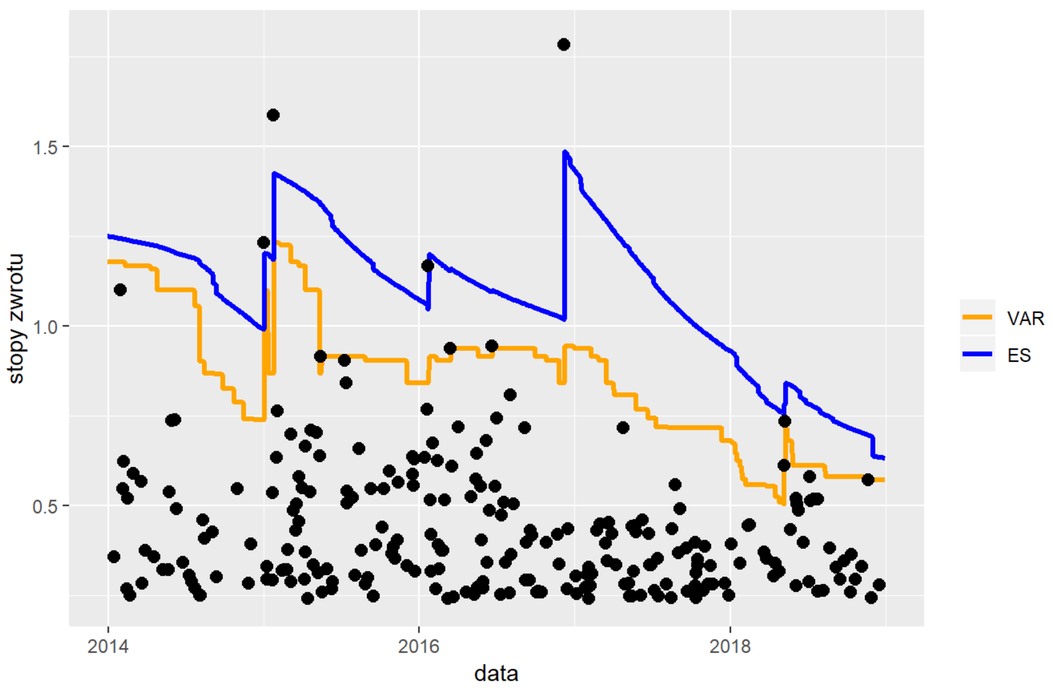
Wynik symulacji historycznej z wagami

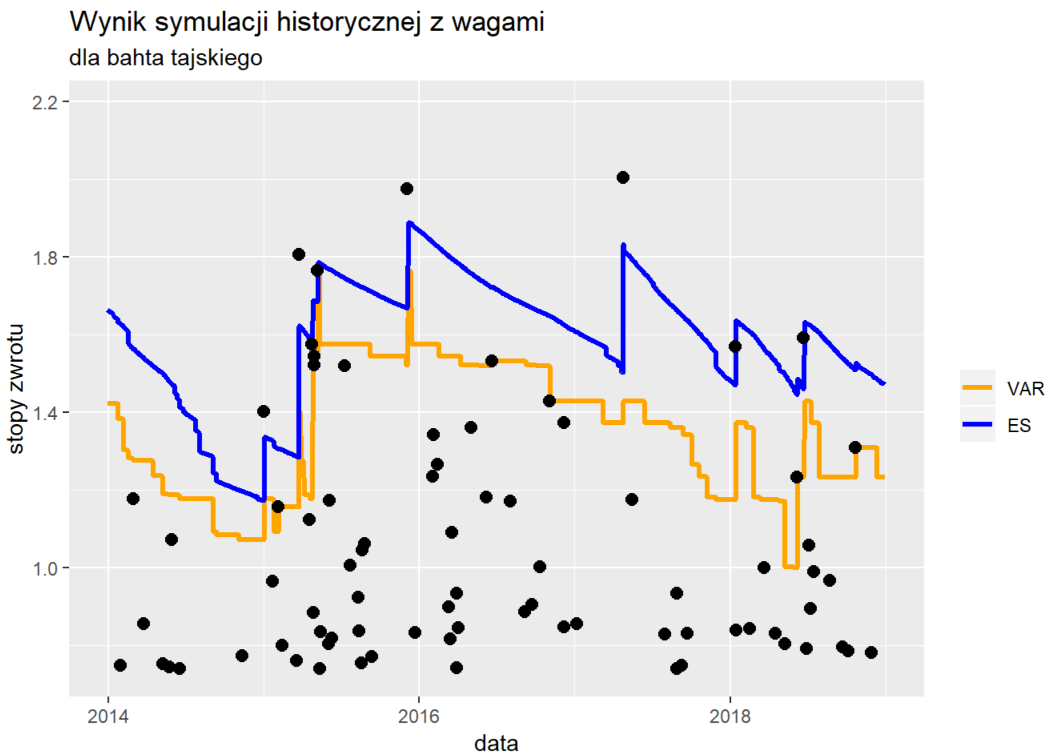
dla funta brytyjskiego



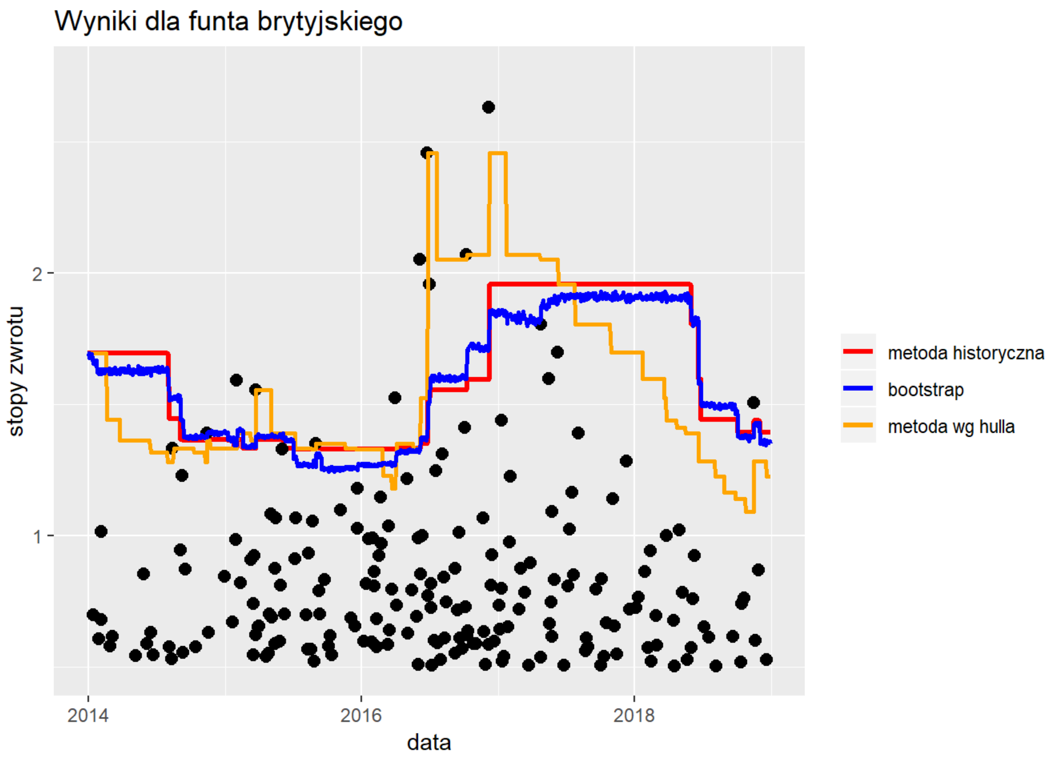
Wynik symulacji historycznej z wagami

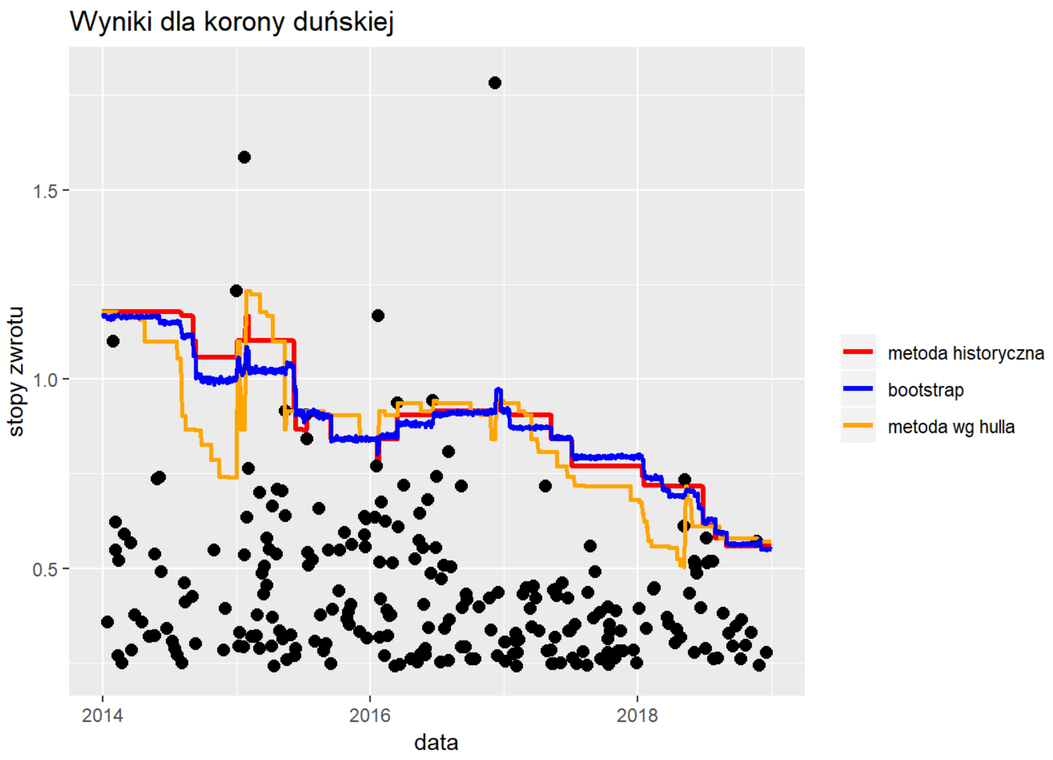
dla korony duńskiej

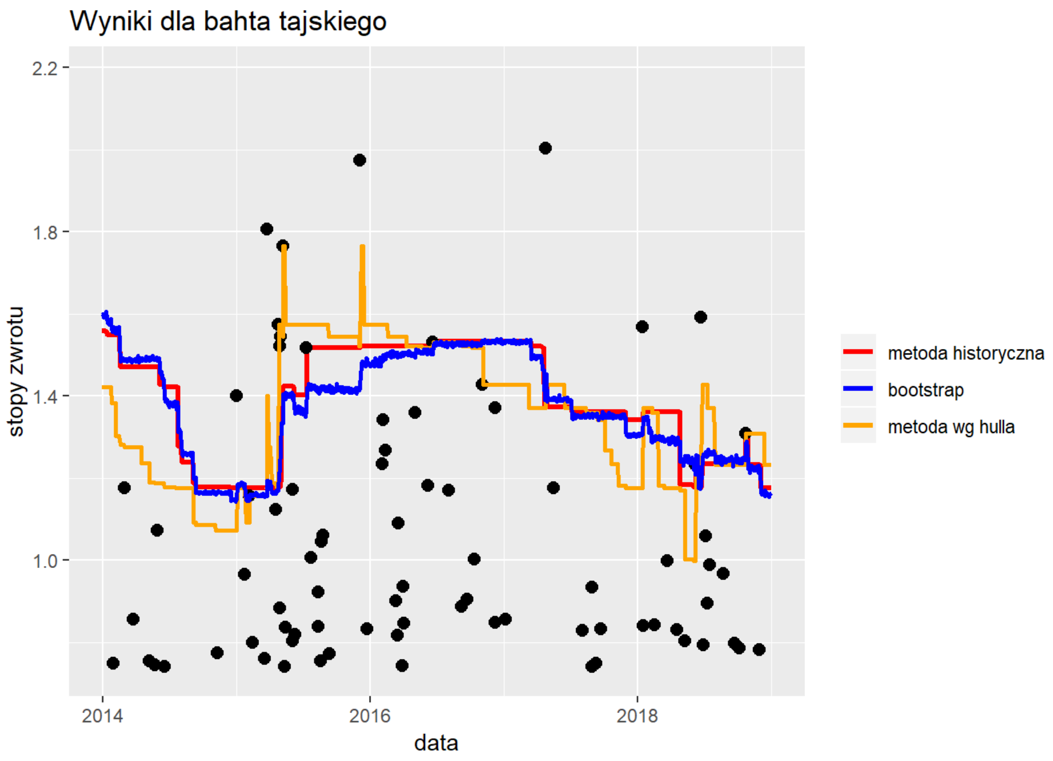




4.4 Wizualne porównanie metod







4.5 Komentarz do wykresów

Na pierwszy rzut oka widać podobne wyniki w przypadku metod symulacji historycznej i bootstrapu - co nie może dziwić ze względu na podobieństwa między tymi metodami. Inaczej sytuacja ma się z metodą historyczną z wagami. Widzimy klarowny wpływ wprowadzonych wag na wartości VaR. Są one coraz mniejsze wraz ze spadkiem wag dla obserwacji przekraczających kwantyl. Gdy do okna 'dołączy' świeża (nowa) obserwacja, która przekracza kwantyl, VaR skacze blisko jej wartości - co spowodowane jest jej dużą w danym oknie estymacji. Zastanawiać może jedynie fakt(w przypadku walut brytyjskiej i tajskiej), dlaczego podczas tych skoków var znajduje się aż tak blisko występującej wtedy wartości straty. Dziwi również, że dla korony duńskiej wykres jest bardziej intuicyjny (brak jest takich wyraźnych skoków tuż pod wartość straty jak w przypadku pozostałych walut).

5 Testy

W celu sprawdzenia, który z testów osiągnął najlepsze wyniki przeprowadzone zostaną testy:

- Kupca,
- Christoffersen'a,

- oraz test wartości rzeczywistych.

Pierwszy z wymienionych testów, test Kupca weryfikuje czy udział przekroczeń w oknach estymacji jest zgodny z zadanym poziomem istotności. Sprawdza ile razy został przekroczony VAR w oknie na podstawie którego został wyznaczony. Na podstawie specjalnej statystyki wyznacza się minimalną i maksymalną liczbę przekroczeń dla danych parametrów (szerokości okna, poziomu istotności). Przykładowo, dla tego badania, (przy 99% VAR i 500 elementowym oknie estymacji) VAR wyznaczony jest poprawnie jeśli w oknie na podstawie którego był estymowany został przekroczony 1,2,..,8,9 lub 10 razy. H0 testu - VaR wyznaczony jest poprawnie.

Test Christoffersen'a sprawdza natomiast czy przekroczenia jakie miały miejsce występują niezależnie w czasie. Nagłe wystąpienie kilku wyjątków w krótkim czasie może oznaczać poważne problemy dla danej instytucji finansowej ze względu na możliwe duże straty poniesione w krótkim okresie, więc rzecz jasna zależy im na tym, aby te przekroczenia były niezależne. H0 testu - przekroczenia VaR są niezależne w czasie.

Ostatni z testów, jak sama nazwa wskazuje, sprawdza jak wyznaczony var sprawdził się w praktyce. Porównujemy wyznaczone wartości var do strat, które miały miejsce w rzeczywistości i zliczamy, ile razy var został przekroczony. Dzielimy sumę przekroczeń przez liczbę dni dla których dokonaliśmy sprawdzenia VaR i otrzymujemy procentowy udział dni, w których został przekroczony var. Im bliższa jest to wartość do poziomu istotności który przyjęliśmy, tym lepiej.

5.1 Wykorzystane testy i ich implementacja

TEST KUPCA

```
test_kupca <- function(straty, var, d = 500){
  res <- NA
  suma_wyj <- NA
  for (i in (1:length(var)))
  {
    okno <- straty[i:(d+i-1)]
    #liczam wystąpienie wyjątków
    temp <- ifelse(okno > var[i], 1 , 0)
    suma_wyj[i] <- sum(temp)
    #sprawdzam czy mieszczą się w odpowiednim przedziale
    #jeśli tak to var wyznaczony jest poprawnie - 1
    if (suma_wyj[i] < 11 && suma_wyj[i] > 0)
    {
      res[i] <- 1
    } else {res[i] <- 0}
  }
  acc <- sum(res)/length(var)
  return(round(acc*100,2))
}
```

TEST CHRISTOFFERSENA

```r
#d jest szerokością okna na podstawie którego wyznaczany jest var
christoffersen_test <- function(straty, var, d = 500){
  res <- NULL
  for (i in 1:length(var)){
    a00 <- 0
    a01 <- 0
    a10 <- 0
    a11 <- 0

    #deklarowanie okna na podstawie ktorego liczony byl dany var
    okno <- straty[i:(d-1+i)]
    wyj <- ifelse(okno > var[i], 1, 0)

    for(n in 1:(d-1)){
      #zliczanie zdarzen potrzebnych do nastepnego etapu
      if(wyj[n] == 0 && wyj[n+1]==0){
        a00 = a00+1
      } else if(wyj[n] == 1 && wyj[n+1]==0){
        a10 = a10+1
      } else if(wyj[n] == 0 && wyj[n+1]==1){
        a01 = a01+1
      } else {
        a11 = a11+1
      }
    }
    #obliczanie niezbędnych do przeprowadzenia testu statystyk
    q0 <- a00 / (a00 + a01)
    q1 <- a10 / (a10 + a11)
    qq <- (a00 + a10) / (a00 + a10 + a01 + a11)
    l_ratio <- -2*log((qq/q0)^a00*((1-qq)/(1-q0))^a01*(qq/q1)^a10*((1-qq)/(1-q1))^a11)

    if ( l_ratio > 6.635 ){
      # 0 jesli odrzucamy h0, 1 jeśli przyjmujemy
      res[i] <- 0
    } else {res[i] <- 1}
  }
  #acc określa jaki jest odsetek poprawnie wyznaczonych VAR
  acc <- sum(res)/length(res)
  return(round(acc*100,2))
}
```

TEST WARTOŚCI RZECZYWISTYCH

```r
twr <- function(var, straty) {
  #zlicza wyjatki
  suma_wyj <- ifelse(var < straty, 1, 0)

  res <- sum(suma_wyj)/length(var)

  return(round(res*100,2))
}
```

5.2 Wyniki testów

Ze względu na specyfikę metody symulacji historycznej, nie ma rzecz jasna sensu przeprowadzania dla niej testu kupca. W przypadku metody bootstrap istnieje niewielka, w praktyce niemal równa 0 szansa, że wynik dla tej metody będzie wskazywał na jakiekolwiek błędy. Musiałoby dojść do szeregu bardzo pechowych losowań, aby to doszło do skutku, co przy liczbie symulacji stosowanych w praktyce jest niemożliwe.

Wyniki podane w tabeli oznaczaja:

- w przypadku testów Kupca i Christoffersen'a - odsetek VARów wyznaczonych poprawnie w świetle wymagań danego testu,
- procent dni w jakie VAR został przekroczony przez wartość straty występująca w danym dniu - w przypadku testu wartości rzeczywistych.

Dla funta brytyjskiego:

	Test.kupca	Test.christoffersena	Test.wartości.rzeczywistych
metoda historczyna	NA	100.00	0.95

	Test.kupca	Test.christoffersena	Test.wartości.rzeczywistych
metoda bootstrap	100.00	100.00	0.95
metoda hulla	99.21	87.25	0.95

Dla korony duńskiej:

	Test.kupca	Test.christoffersena	Test.wartości.rzeczywistych
metoda historczyna	NA	100	0.71
metoda bootstrap	100.00	100	0.71
metoda hulla	93.27	100	0.71

Dla bahta tajskiego:

	Test.kupca	Test.christoffersena	Test.wartości.rzeczywistych
metoda historczyna	NA	100	1.11
metoda bootstrap	100.00	100	1.19
metoda hulla	93.11	100	0.95

Widzimy, że wyniki wszystkich testów są bardzo podobne. W przypadku testu kupca dla bootstrapu moje przypuszczenia okazały się zgodne z rzeczywistością - 100% poprawność wyznaczonych VAR. Zgodnie z wynikami testu Christoffersena widzimy również, że wyłącznie metoda symulacji historycznej z wagami w przypadku funta w 13% wyznaczonych VARow nie spełniła odpowiednich wymagań i nie została zaklasyfikowana jako poprawna pod względem niezależności wyjątków w czasie. Test wartości rzeczywistych w przypadku walut GBP i DKK wskazuje na te same wyniki, natomiast dla bata najlepsza wydaje się metoda hulla, a co ciekawe bootstrap okazał się gorszy od metody historycznej.

6. Podsumowanie

Wnioski z badania:

- na podstawie testów ciężko stwierdzić, która z metod daje najlepszy wynik,
- metoda symulacji historycznej z wagami patrząc pod kątem wyników christoffersena jako jedyna z wykorzystanych metoda nie ma 100% poprawności w przypadku każdej z walut,
- jednak już na podstawie testu wartości rzeczywistych wydaje się być najlepszą z zastosowanych w tym badaniu metod,
- metody symulacji historycznej i bootstrapu uzyskują bardzo zbliżone wyniki, a co ciekawe ciut lepsza okazała się ta prostszą z metod. Zasadne wydaje się w takim razie pytanie, czy skoro dają podobne wyniki to jest sens stosować metodą zdecydowanie bardziej złożoną obliczeniowo w praktyce ?
- biorąc jednak pod uwagę wykresy jakie przedstawione były w trakcie badania, to właśnie metoda bootstrapu moim zdaniem wygląda najbardziej sensownie,
- **co z tego wynika ? Tylko i aż tyle, że ciężko jednoznacznie wybrać, która z metod okazała się najlepsza. Każda ma swoje [illegible]y, na podstawie których ciężko zdecydowanie wyróżnić 1 najlepszą.**