

# Estymacja prawdopodobieństwa sukcesu

Aleksandra Palka

13 listopada 2023

## 1 Wstęp

Praca skupiona jest na metodach estymacji prawdopodobieństwa sukcesu  $p$  w rozkładzie dwumianowym  $B(n, p)$ . Porównane zostaną różne przedziały ufności dla parametru  $p$ . Omówione będą też testy do weryfikacji hipotez dotyczących prawdopodobieństwa sukcesu - dokładne i asymptotyczne.

## 2 Zadanie 1 - porównanie przedziałów ufności

Przeprowadzimy symulacje, których celem jest porównanie prawdopodobieństwa pokrycia i długości przedziałów ufności Cloppera-Pearsona (metoda *exact*), Walda (metoda *asymptotic*) i Bayesa zaimplementowanego w funkcji *binom.confint* pakietu *binom*. Uwzględnimy poziom ufności 0.95, rozmiary próby  $n \in \{30, 100, 1000\}$  i wartości prawdopodobieństwa  $p \in \{0.01, 0.02, \dots, 0.99\}$ . Kroki symulacji:

1. Gnerujemy  $x \sim B(n, p)$
2. Wyznaczamy przedział ufności dla  $p$  dla wybranej metody korzystając z *binom.confint*.
3. Sprawdzamy czy  $p$  należy do wyznaczonego przedziału ufności./Liczymy długość przedziału ufności.
4. Kroki 1° – 3° powtarzamy  $N$  razy dla ustalonego  $p, n$ .
5. Wyliczamy procent pokrycia =  $\frac{\#\{p \text{ należy do przedziału ufności}\}}{N}$  i średnią długość przedziału ufności.
6. Kroki 1° – 5° powtarzamy dla  $p \in \{0.01, 0.02, \dots, 0.99\}$ .
7. Kroki 1° – 6° powtarzamy dla  $n \in \{30, 100, 1000\}$ .

W realizacji symulacji użyjemy  $N = 500$ . Kod programu:

```
library(binom)
procent_pokrycia_ = function(p, n, metoda, M) {
  count <- 0
  x <- rbinom(M,n,p)
  przedzialy_l <- binom.confint(x,n,methods=metoda)[5]
  przedzialy_u <- binom.confint(x,n,methods=metoda)[6]

  return (sum((p >= przedzialy_l) & (p<=przedzialy_u))/M)
}

procent_pokrycia = function(n, M=500){
  metody = c("exact", "asymptotic", "bayes")
  P = seq(0.01,0.99,by = 0.01)
  pokrycie = matrix(0,length(P),3)
  for (m in 1:3) {
    for (p in 1:length(P)){
      pokrycie[p,m] <- procent_pokrycia_(P[p],n, metody[m],M=M)
    }
  }
}
```

```

    }
  }
  return (pokrycie)
}

dlugosc_przedzialu_ = function(p, n, metoda, M) {
  x <- rbinom(M,n,p)
  przedzialy_l <- binom.confint(x,n,methods=metoda)[5]
  przedzialy_u <- binom.confint(x,n,methods=metoda)[6]
  return (mean((przedzialy_u-przedzialy_l)[,1]))
}

dlugosc_przedzialu = function(n, M=500){
  metody = c("exact", "asymptotic", "bayes")
  P = seq(0.01,0.99,by = 0.01)
  przedzialy = matrix(0,length(P),3)
  for (m in 1:3) {
    for (p in 1:length(P)){
      przedzialy[p,m] <- dlugosc_przedzialu_(P[p],n, metody[m],M=M)
    }
  }
  return (przedzialy)
}

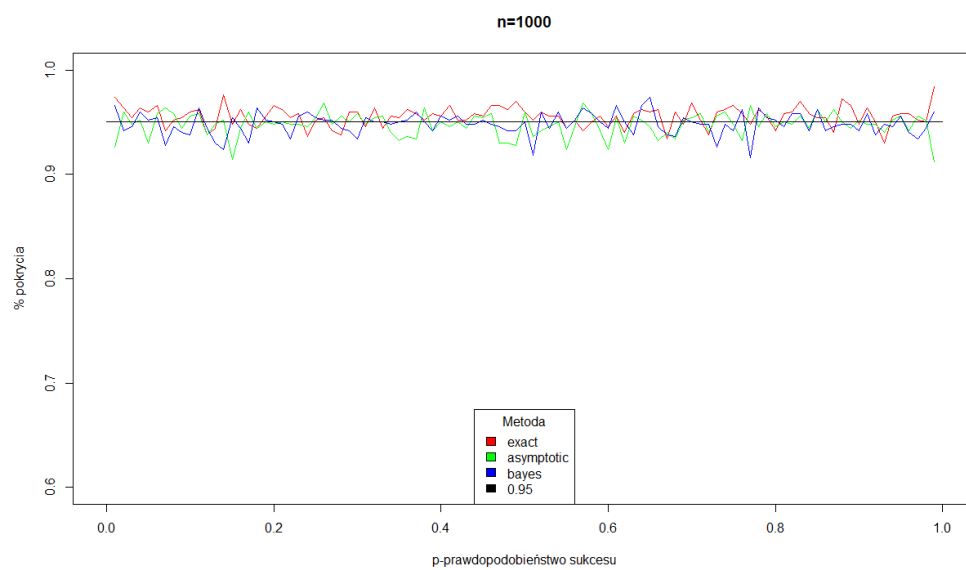
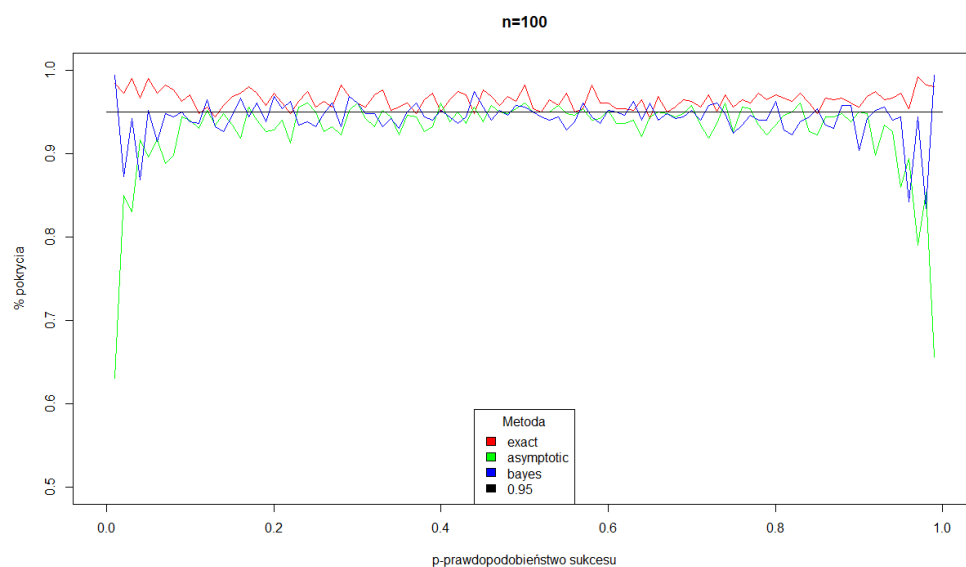
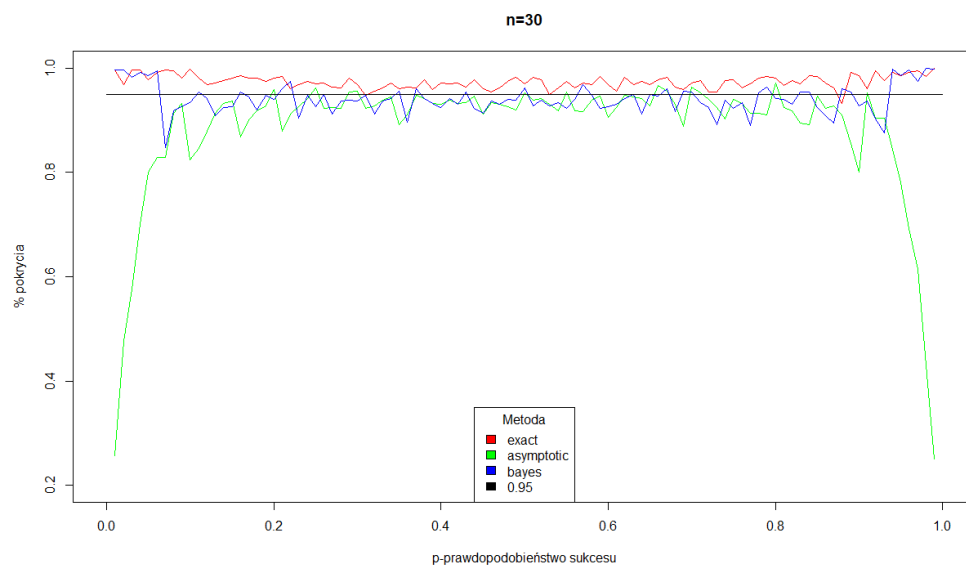
P = seq(0.01,0.99,by = 0.01)
pokrycie30 = procent_pokrycia(30)
pokrycie100 = procent_pokrycia(100)
pokrycie1000 = procent_pokrycia(1000)

przedzialy30 = dlugosc_przedzialu(30)
przedzialy100 = dlugosc_przedzialu(100)
przedzialy1000 = dlugosc_przedzialu(1000)

```

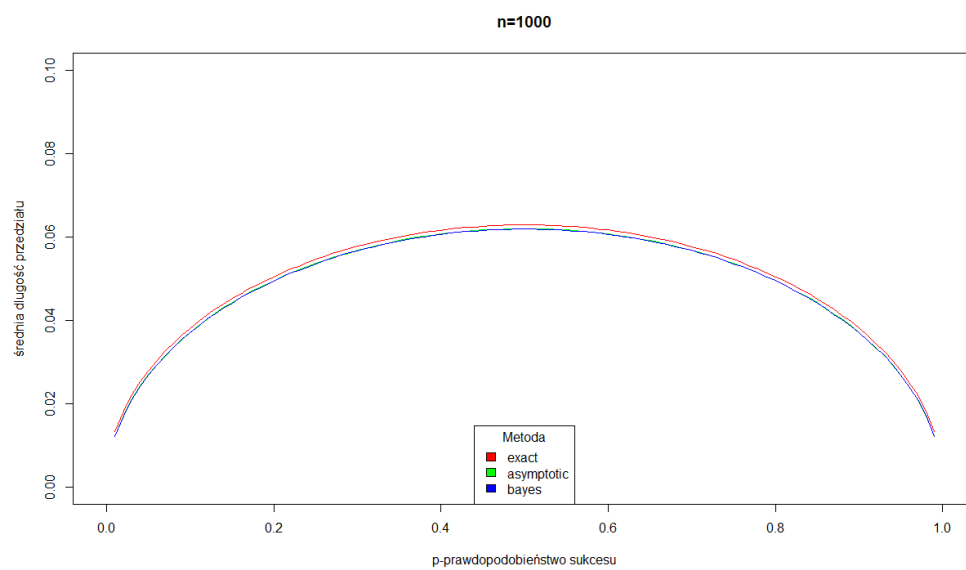
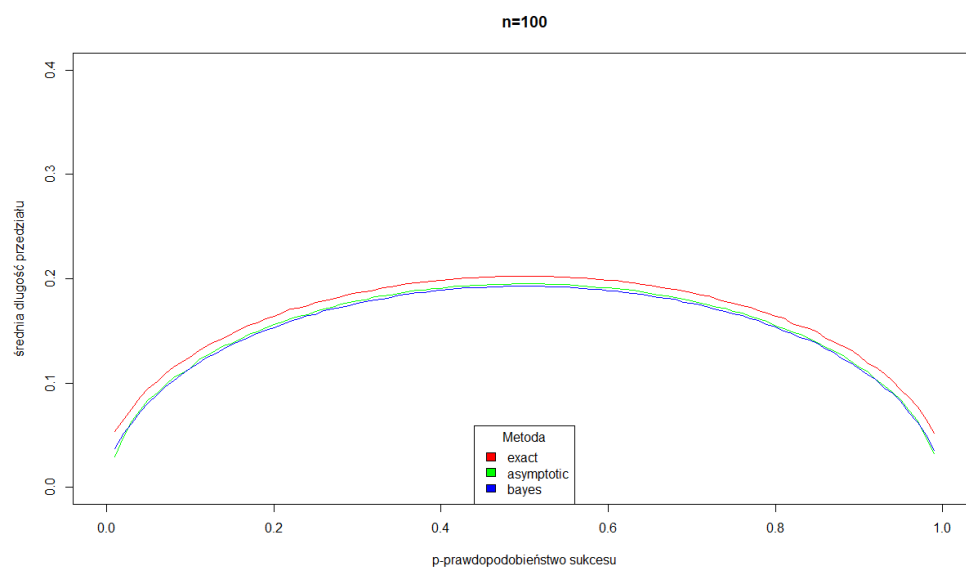
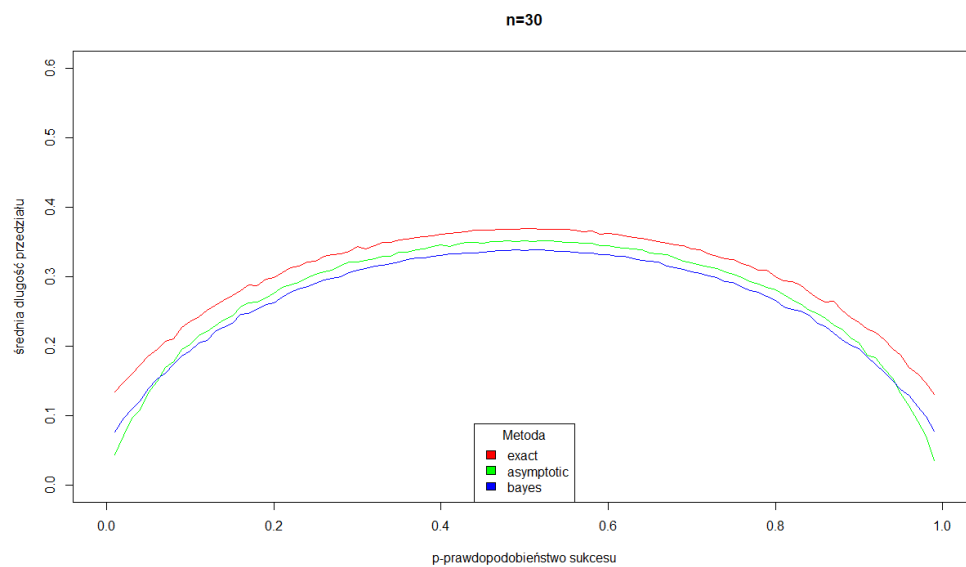
Prawdopodobieństwa pokrycia, czyli prawdopodobieństwa, że  $p$  użyte do wygenerowania realizacji znajduje się w wyznaczonym przedziale ufności przedstawione są na rys. 1. Możemy zauważyć, że rozmiar próby  $n$  wpływa na ich wartości. Dla większego  $n$  różnice między tymi prawdopodobieństwami wyznaczonymi różnymi metodami są najmniejsze, podczas, gdy dla małego  $n$  są one czasami bardzo duże. Do tego dla dużego  $n$  wartości te oscylują najbliżej spodziewanej wartości 0.95, mają najmniejsze wahania i nie można wskazać oczywistej różnicy między użytymi metodami. Dla  $n = 30$  i  $n = 100$  możemy zauważyć, że dla skrajnych wartości  $p$  wykorzystane asymptotyczne przedziały ufności prowadzą do znacznie mniejszych wartości % pokrycia od innych testów, znacznie różniących się od 0.95. Zatem dla małych  $n$  i  $p$  przedziały Walda są niedokładne i nie są najlepszym wyborem. Pozostałe przedziały również dla skrajnych  $p$  mają wyniki inne od reszty, lecz nie są one aż tak różne. Metoda dokładna prowadzi do wartości % pokrycia nieznacznie większych od 0.95 i pozostałych metod, jej przedziały ufności są mniej konserwatywne.

Długości przedziałów ufności (rys. 2) potwierdzają te spostrzeżenia. Dla małych  $n$  przedziały dokładne są najdłuższe, następnie przedziały Bayesa, a najkrótsze są asymptotyczne. Jedynie dla skrajnych  $p$  Bayes i asymptotyczne zachowują się odwrotnie. Wraz ze wzrostem  $n$  przedziały ufności stają się krótsze, a różnice pomiędzy poszczególnymi metodami stają się nieistotne. Zatem, dla dużych rozmiarów próby podobne wyniki uzyskamy każdą z metod. Inaczej jest dla mniejszych rozmiarów i małych prawdopodobieństw sukcesu, gdzie widać niedokładność metody asymptotycznej. Z kolei metoda dokładna, ponieważ jej przedziały są najszerze, będzie dla małych  $n$  i nie skrajnych  $p$  najmniej precyzyjna.



3

Rysunek 1: Prawdopodobieństwa pokrycia dla różnych n.



### 3 Zadanie 2

Na podstawie danych zawartych w pliku Choroba.csv, wyznaczamy realizacje przedziałów ufności, na poziomie ufności 0.95, dla prawdopodobieństwa, że losowo wybrana osoba z badanej populacji jest chora.

Korzystamy z funkcji `tbinom.confint(x,n,conf.level,methods)` (gdzie  $x$ -realizacja zmiennej z rozkładu dwumianowego (liczba chorych respondentów, czyli 56),  $n$ -ilość prób (liczba respondentów, czyli 196), *conf.level*-poziom ufności, *methods*-metoda), która zwraca przedział ufności  $p$  dla podanej realizacji.

Wyznaczamy realizacje wszystkich możliwych typów tych przedziałów, porównujemy te realizacje i ich długości.

	Metoda	Dolna	Gorna	Długosc
1	exact	0.2236035	0.3544347	0.1308311
2	ac	0.2269321	0.3527347	0.1258026
3	asymptotic	0.2224699	0.3489587	0.1264888
4	wilson	0.2270645	0.3526023	0.1255378
5	prop.test	0.2247279	0.3552673	0.1305394
6	bayes	0.2245686	0.3502467	0.1256781
7	logit	0.2268493	0.3528824	0.1260331
8	cloglog	0.2242758	0.3499864	0.1257106
9	probit	0.2260215	0.3520157	0.1259942

Rysunek 3: Przedziały ufności dla prawdopodobieństwa z zadania 5 wyznaczone różnymi metodami

Wnioski: W badaniu wzięło udział 196 osób, więc  $n$  jest dość duże. Oszacowane  $p$  nie jest też wartością skrajną. Zatem na podstawie zadania 1, jeśli zależy nam na jak precyzji, tj. krótszym przedziale, możemy skorzystać z np. przedziałów Walda. Użycie przedziałów Cloppera-Pearsona prowadziłoby do mniejszej dokładności.

### 4 Zadanie 3

Dla danych z pliku Choroba, na poziomie istotności 0.05 zweryfikujemy hipotezy:

- a) prawdopodobieństwo, że losowo wybrana osoba z badanej populacji jest chora jest mniejsze bądź równe  $\frac{1}{2}$ :
  - $p_0$ -prawdopodobieństwo, że losowo wybrana osoba jest chora .
  - Hipoteza zerowa  $H_0 : p_0 \leq \frac{1}{2}$ .
  - Hipoteza alternatywna  $H_1 : p_0 > \frac{1}{2}$ .
  - Wartość poziomu krytycznego:
    - \* z `prop.test(correct=TRUE)`: 1,
    - \* z `prop.test(correct=FALSE)`: 1,
    - \* z `binom.test`: 1.
  - Weryfikacja hipotezy: przyjmujemy hipotezę zerową.
- b) prawdopodobieństwo, że losowo wybrana osoba z sektora 1 jest chora jest równe prawdopodobieństwu, że losowo wybrana osoba z sektora 2 jest chora:
  - $p_1$ -prawdopodobieństwo, że osoba z sektora 1. jest chora,
  - $p_2$ -prawdopodobieństwo, że osoba z sektora 2. jest chora.
  - Hipoteza zerowa  $H_0 : p_1 = p_2$ .
  - Hipoteza alternatywna  $H_1 : p_0 \neq p_2$ .
  - Wartość poziomu krytycznego:

- \* z  $prop.test(correct=TRUE)$ : 0.0004271 ( $< 0.05$ ),
  - \* z  $prop.test(correct=FALSE)$ : 0.0002297 ( $< 0.05$ ).
  - o Weryfikacja hipotezy: odrzucamy hipotezę zerową .
- c) prawdopodobieństwo, że losowo wybrana osoba z badanej populacji jest chora jest mniejsze bądź równe  $\frac{1}{2}$  dla osoby o średnim statusie ekonomicznym :
- o  $p_0$ -prawdopodobieństwo, że losowo wybrana osoba o średnim statusie ekonomicznym jest chora.
  - o Hipoteza zerowa  $H_0 : p_0 \leq \frac{1}{2}$ .
  - o Hipoteza alternatywna  $H_1 : p_0 > \frac{1}{2}$ .
  - o Wartość poziomu krytycznego z  $binom.test$ : 0.9998.
  - o Weryfikacja hipotezy: przyjmujemy hipotezę zerową
- d) prawdopodobieństwo, że losowo wybrana osoba z sektora 1 jest chora jest równe prawdopodobieństwu, że losowo wybrana osoba z sektora 2 jest chora dla osoby o średnim statusie ekonomicznym:
- o  $p_1$ -prawdopodobieństwo, że osoba o średnim statusie ekonomicznym z sektora 1. jest chora,
  - o  $p_2$ -prawdopodobieństwo, że osoba o średnim statusie ekonomicznym z sektora 2. jest chora.
  - o Hipoteza zerowa  $H_0 : p_1 = p_2$ .
  - o Hipoteza alternatywna  $H_1 : p_0 \neq p_2$ .
  - o Wartość poziomu krytycznego dla b:
    - \*  $prop.test(correct=TRUE)$ : 0.02758608 ( $< 0.05$ ),
    - \*  $prop.test(correct=FALSE)$ : 0.01149 ( $< 0.05$ ).
  - o Weryfikacja hipotezy: odrzucamy hipotezę zerową.

## 5 Zadanie 4

Hipoteza zerowa to  $H_0 : p_0 = 0.5$ , przy hipotezie alternatywnej  $H_1 : \neq 0.5_0$  na poziomie istotności  $\alpha = 0.05$ , gdzie  $p$  jest prawdopodobieństwem sukcesu w rozkładzie dwumianowym. Dla tak sformułowanych hipotez oszacujemy symulacyjnie prawdopodobieństwo błędu I-go rodzaju dla testu dokładnego i asymptotycznego oraz porównamy moce tych testów.

### 5.1 Zadanie 4a) - szacowanie prawdopodobieństwa błędu I-go rodzaju

Błąd I-go rodzaju to odrzucenie prawdziwej hipotezy zerowej. Jest on bliski poziomowi istotności  $\alpha$ . Przeprowadzamy symulację, aby oszacować prawdopodobieństwo błędu I-go rodzaju testu dokładnego i testu asymptotycznego, przy przyjętym poziomie istotności  $\alpha$  dla rozmiaru próby  $n \in \{30, 100, 1000\}$ . W tym celu:

1. Generujemy dane spełniające  $H_0$ , czyli  $x \sim B(n, 0.5)$ .
2. Korzystając z funkcji  $binom.test$ ,  $prop.test$  z poprawką i bez poprawki przeprowadzamy test.
3. Sprawdzamy czy  $H_0$  została odrzucona ( $p - value < 0.05$ ).
4. Powtarzamy kroki 1° – 3°  $N$  razy.
5. Błąd I rodzaju to  $\frac{\#\{H_0 \text{ odrzucona}\}}{N}$ .

Dla wszystkich symulacji przyjmujemy liczbę powtórzeń  $N = 5000$ . Kod wykorzystany do symulacji:

```

binom.test_p = function(x,n,p=0.5, alternative, conf.level=0.95){
  return (unlist((binom.test(x,n,p=0.5, alternative="t", conf.level=conf.level)$p.value)))
} #pomocnicza funkcja
prop.test_p = function(x,n,p=0.5, alternative, conf.level=0.95, correct=TRUE){
  return (unlist((prop.test(x,n,p=0.5, alternative="t", conf.level=conf.level, correct=correct)$p.value)))
} #pomocnicza funkcja

blad1rodzaju = function(n,N=5000,conf=0.95){
  x = rbinom(N, n, 0.5)
  #liczenie p-wartosci dla poszczegolnych realizacji

  p1_ <- lapply(x, FUN=binom.test_p, n=n, p=0.5, alternative="t", conf.level=conf)
  p2_ <- lapply(x, FUN=prop.test_p, n=n, p=0.5, alternative="t", conf.level=conf, correct=TRUE)
  p3_ <- lapply(x, FUN=prop.test_p, n=n, p=0.5, alternative="t", conf.level=conf, correct=FALSE)

  #sprawdzanie czy p-wartosc <0.05
  p1 <- sum(p1_<1-conf)/N
  p2 <- sum(p2_<1-conf)/N
  p3 <- sum(p3_<1-conf)/N

  return (c(p1, p2,p3))
}
blad1_30 = blad1rodzaju(30)
blad1_100 = blad1rodzaju(100)
blad1_1000 = blad1rodzaju(1000)

```

$n$	<i>binom.test</i>	<i>prop.test</i> z poprawką	<i>prop.test</i> bez poprawki
30	0.0422	0.0422	0.0422
100	0.0374	0.0374	0.0588
1000	0.0476	0.0476	0.0550

Wartości błędy I-go rodzaju testu dokładnego i asymptotycznego z poprawką na ciągłość dla sprawdzanych  $n$  są takie same. Różnice można zauważyć porównując je do testu asymptotycznego bez poprawki dla  $n \in \{100, 1000\}$ . Błąd I-go rodzaju ma dla tego testu większe wartości. Przeprowadzanie testu asymptotycznego bez poprawki może częściej prowadzić do odrzucenia prawdziwej hipotezy zerowej.

## 5.2 Zadanie 4b) - porównanie mocy testów

Moc testu to prawdopodobieństwo odrzucenia fałszywej hipotezy zerowej. Porównamy moce testu asymptotycznego i dokładnego przy przyjętym poziomie istotności  $\alpha$  dla rozmiaru próby  $n \in \{30, 100, 1000\}$  przeprowadzając symulację:

1. Generujemy dane nie spełniające  $H_0(p \neq 0.5)$ , czyli  $x \sim B(n, p)$
2. Korzystając z funkcji *binom.test*, *prop.test* z poprawką i bez poprawki przeprowadzamy test.
3. Sprawdzamy czy  $H_0$  została odrzucona ( $p - value < 0.05$ ).
4. Powtarzamy kroki 1° – 3°  $N$  razy.
5. Moc testu to  $\frac{\#\{H_0 \text{ odrzucona}\}}{N}$ .
6. Kroki 1° – 5° powtarzamy dla  $p \in \{0.01, 0.02, \dots, 0.49, 0.51, 0.52, \dots, 0.99\}$ .

Dla wszystkich symulacji przyjmujemy liczbę powtórzeń  $N = 500$ . Kod do przeprowadzenia symulacji:

```

binom.test_p = function(x,n,p=0.5, alternative, conf.level=0.95){
  return (unlist((binom.test(x,n,p=0.5, alternative="t", conf.level=conf.level)$p.value)))
}

```

```

prop.test_p = function(x,n,p=0.5, alternative, conf.level=0.95, correct=TRUE){
  return (unlist((prop.test(x,n,p=0.5, alternative="t", conf.level=conf.level, correct=correct)$p.value)))
}
moctestu_ = function(p,n,N,conf=0.95){ #dla ustalonego p
  x = rbinom(N, n, p)
  p1_ <- lapply(x, FUN=binom.test_p, n=n, p=0.5, alternative="t", conf.level=conf)
  p2_ <- lapply(x, FUN=prop.test_p, n=n, p=0.5, alternative="t", conf.level=conf, correct=TRUE)
  p3_ <- lapply(x, FUN=prop.test_p, n=n, p=0.5, alternative="t", conf.level=conf, correct=FALSE)

  p1 <- sum(p1_<0.05)/N
  p2 <- sum(p2_<0.05)/N
  p3 <- sum(p3_<0.05)/N

  return (c(p1,p2,p3))
}

moctestu = function(n, N=500,conf=0.95){ # dla wielu p
  P = c(seq(0.01,0.49,by = 0.01),seq(0.51,0.99,by = 0.01))
  blad = matrix(0,length(P),3)
  for (p in 1:length(P)){
    blad[p,] <- moctestu_(P[p],n, N=N,conf = conf)
  }
  return (blad)
}
blad30 = moctestu(30,N=500)
blad100 = moctestu(100,N=500)
blad1000 = moctestu(1000,N=500)

```

Na rysunku 4 przedstawione są moce testów oszacowane w tej symulacji. Ogólnie, wraz ze zbliżaniem się  $p$  do  $p_0$  moc testu maleje i zbliża się do wartości poziomu istotności, czyli 0.05. Im większy jest rozmiar próby tym później i gwałtowniej to następuje.

Dla  $n$  widzimy, że użyty test nie wpłynął na moc testu, jest ona taka sama dla każdego rodzaju.

Inaczej wygląda to dla  $n = 100$  i  $n = 1000$ . Moce testów dokładnego i asymptotycznego z poprawką pokrywają się, ale różnią od testu asymptotycznego bez poprawki. Dla niego od momentu zmiany moc jest nieznacznie, acz zauważalnie większa. Oznacza to, że prawdopodobieństwo popełnienia błędu II-go rodzaju przyjęcia fałszywej hipotezy zerowej jest dla tego testu większe.



