Podstawy rachunku prawdopodobieństwa i statystyki w R

Uniwersytet Marii Curie-Skłodowskiej
Wydział Matematyki, Fizyki i Informatyki
Instytut Informatyki i Matematyki

Podstawy rachunku prawdopodobieństwa i statystyki w R

Lublin 2025

Instytut Informatyki i Matematyki UMCS
Lublin 2025

Andrzej Krajka

Podstawy rachunku prawdopodobieństwa i statystyki w R

Spis treści

Rozdział 1

Podstawy pracy z R

1.1. Przygotowanie danych

Dane można wczytac albo z internetu (*dane*), albo przygotowane z twardego dysku, albo samemu przygotować (tutaj wektor x i ramka danych *dane1*). Zwróć uwagę na to, że wszystkie elementy wektora muszą być tego samego typu (***int, char, logi, factor, ordered factor***) a wszystkie elementy każdej kolumny ramki danych też muszą być tego samego typu, chociaż sąsiednie kolumny mogą już mieć inny typ.

```
> x<-c(1.2,1.2,3.5,5.5,4.9,3.3,2.2,1.22)
> dane<-read.csv("http://www.biecek.pl/R/dane/daneSoc.csv", sep=";")
> dane1<-data.frame(Prog=c("Java","Python","R","R","C++","C#"),
+                   time=c(3.15,4.21,5.01, 3.8, 1.2, 1.4))
> str(x)

 num [1:8] 1.2 1.2 3.5 5.5 4.9 3.3 2.2 1.22

> str(dane)

'data.frame':        204 obs. of  7 variables:
 $ wiek                 : int  70 66 71 57 45 48 53 61 63 47 ...
 $ wyksztalcenie        : chr  "zawodowe" "zawodowe" "zawodowe" "srednie" ..
 $ st.cywilny           : chr  "w zwiazku" "w zwiazku" "singiel" "w zwiazku"
 $ plec                 : chr  "mezczyzna" "kobieta" "kobieta" "mezczyzna" .
 $ praca                : chr  "uczen lub pracuje" "uczen lub pracuje" "ucze
 $ cisnienie.skurczowe  : int  143 123 167 150 130 138 149 103 125 158 ...
 $ cisnienie.rozkurczowe: int  83 80 80 87 83 75 80 63 78 89 ...

> str(dane1)

'data.frame':        6 obs. of  2 variables:
 $ Prog: chr  "Java" "Python" "R" "R" ...
 $ time: num  3.15 4.21 5.01 3.8 1.2 1.4
```

1.2. Statystyki pozycyjne

Większość statystyk znajduje się w standartowo wczytywanych bibliotekach *stats* i *base*, tylko skośność i kurtoza jest w bibliotece *e1071*.

```
> library(e1071)
> y<-c(max(x),min(x),mean(x),median(x),IQR(x),var(x),sd(x),mad(x),
+      kurtosis(x),skewness(x))
> y

 [1]  5.500000  1.200000  2.877500  2.750000  2.635000  2.909764  1.705803
 [8]  2.283204 -1.703602  0.310274
```

```
> summary(x)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.200   1.215   2.750   2.877   3.850   5.500

> table(x)

x
 1.2 1.22  2.2  3.3  3.5  4.9  5.5
   2    1    1    1    1    1    1
```

Nie wymieniona tutaj funkcja *quantile(x,p)* oblicza wektor kwantyli rzędu p (p może być liczbą lub wektorem) z wektora x. Ponadto mamy

$$\begin{aligned} IRQ(x) &= quantile(x, \frac{3}{4}) - quantile(x, \frac{1}{4}), \\ mad(x) &= 1.4826 \cdot median(abs(x - median(x))), \\ kurtosis(x) &= \frac{n \sum_{i=1}^{n} (x_i - mean(x))^4}{(\sum_{i=1}^{n} (x_i - mean(x))^2)^2} - 3, \\ skewness(x) &= \frac{\sqrt{n} \sum_{i=1}^{n} (x_i - mean(x))^3}{(\sum_{i=1}^{n} (x_i - mean(x))^2)^{\frac{3}{2}}}, \end{aligned}$$

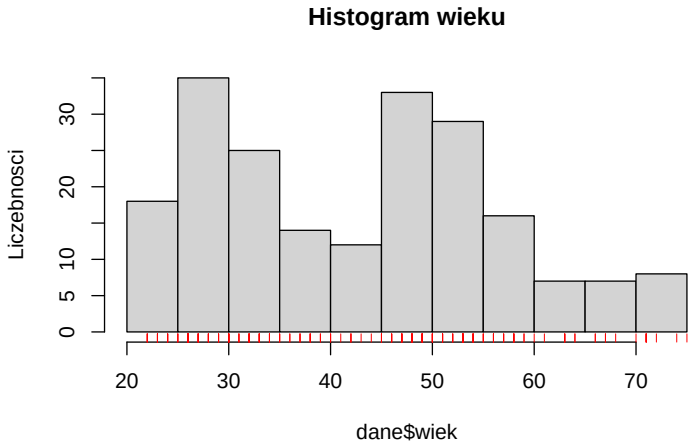
```
> dane$wyksztalcenie<-factor(dane[,2], levels=c("podstawowe","zawodowe",
+                                               "srednie","wyzsze"), ordered
> dane[,4]<-as.factor(dane[,4])
> head(dane)

  wiek wyksztalcenie st.cywilny      plec             praca cisnienie.skurcz
1   70      zawodowe  w zwiazku mezczyzna uczen lub pracuje
2   66      zawodowe  w zwiazku   kobieta uczen lub pracuje
3   71      zawodowe    singiel   kobieta uczen lub pracuje
4   57       srednie  w zwiazku mezczyzna uczen lub pracuje
5   45       srednie  w zwiazku   kobieta uczen lub pracuje
6   48       srednie  w zwiazku mezczyzna       nie pracuje
  cisnienie.rozkurczowe
1                    83
2                    80
3                    80
4                    87
5                    83
6                    75
```

1.3. Podstawowe wykresy

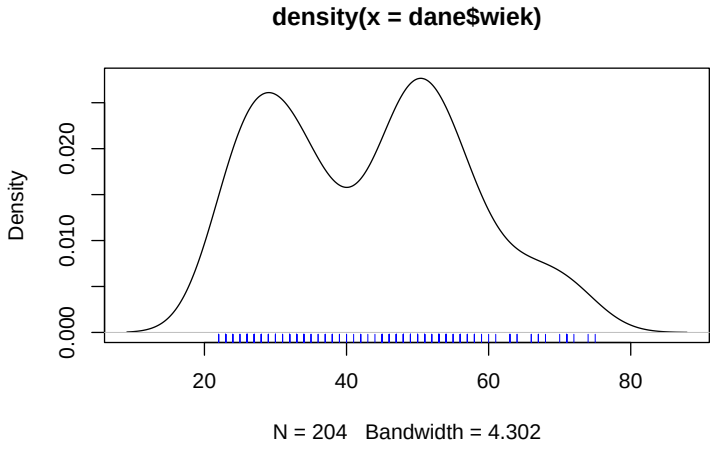
1.3.1. Wykres typu *boxplot*

```
> hist(dane$wiek,10,main="Histogram wieku", ylab="Liczebnosci")
> rug(dane$wiek,side=1, ticksize=0.03, col="red")
```

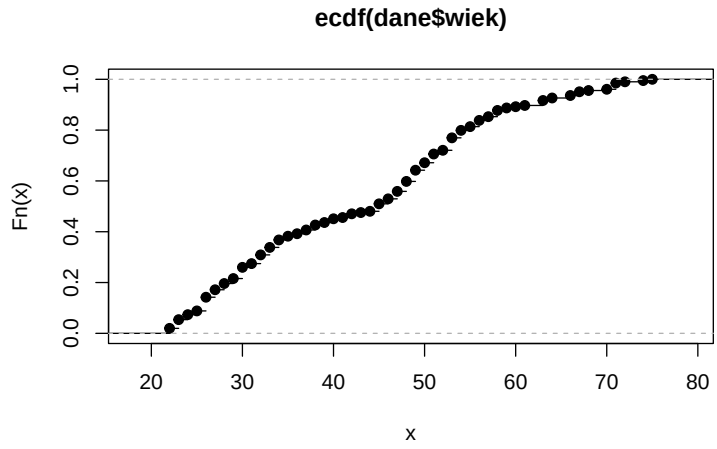


1.3.2. Wykres wygładzonej gęstości i dystrybuanty

```
> plot(density(dane$wiek))
> rug(dane$wiek,side=1, ticksize=0.03, col="blue")
```




```
> plot(ecdf(dane$wiek))
```



1.4. Generator liczb pseudolosowych i rozkłady

1.4.1. Genertatory liczb losowych

```
> ? RNGkind
> .Random.seed[1:20]

 [1]       10403          10   680233772   423961391   593959397   -81247076
 [7] -1626165006  1223428461  1113089703  -739915570  -363985368   704712235
[13] -1665104823  1793078616  1457531910 -1214624287   795801459  1103287266
[19]   294697812  1904416343

> rnorm(10,mean=-1,sd=2)

 [1] -3.5628388 -2.9085888 -3.4439390  1.1669270  2.5362629 -3.2862220
 [7]  0.5382683  0.1994519 -3.7594606 -0.5695365

> punif(q=0.2)

[1] 0.2

> dexp(0.05)

[1] 0.9512294

> qt(p=0.995,df=3)

[1] 5.840909
```

Aktualne rodzaje generatorów w języku R to:

Wichmann-Hill Ziarno jest wektorem całkowitym o długości 3 a cykl ma długość 6.9536×10^{12}.

Marsaglia-Multicarry Uzywany jest tutaj RNG wielokrotny-z-przenoszeniem, zgodnie z zaleceniami George'a Marsaglia a okres ma większy niż 2^{60}. Ziarno stanowią dwie liczby calkowite (wszystkie wartosci sa dozwolone).

Super-Duper Słynny Super-Duper firmy Marsaglia z lat 70-tych. Ma okres $\approx 4.6 \times 10^{18}$ dla większości początkowych ziaren. Ziarno składa się z dwóch liczb całlkowitych (druga musi być nieparzysta).

Mersenne-Twister Autorami są Matsumoto i Nishimura (1998). Ziarnem jest 624-wymiarowy zestaw 32-bitowych liczb całkowitych plus bieżąca pozycja w tym zestawie. Okres wynosi $2^{19937}-1$. R używa własnej metody inicjalizacji.

Knuth-TAOCP-2002 32-bitowa wartość całkowita GFSR wykorzystująca opóźniony ciąg Fibonacciego z odejmowaniem:

$$X_j \equiv (X_{j-100} - X_{j-37}) \bmod 2^{30}].$$

a ziarno to zbiór 100 ostatnich liczb. Okres wynosi około 2^{129} .

Knuth-TAOCP Wczesniejsza wersja z Knutha (1997).

L'Ecuyer-CMRG *Kombinowany generator wielokrotności rekurencji* od L'Ecuyer (1999). Okres wynosi okolo 2^{191}. Ziarno to 6 32-bitowych liczby całkowitych bez znaku.

user-supplied Generator dostarczony przez użytkownika.

1.4.2. Rozkłady prawdopodobieństwa

Pierwsza litera oznacza:

r - oznacza wygeneruj próbkę o danym rozkładzie,

p - odczytaj warość dystrybuanty,

d - odczytaj wartość gęstości dla zadanego x

q -odczytaj dla jakiej wartości x dystrubuanta przyjmuje zadaną wartość.

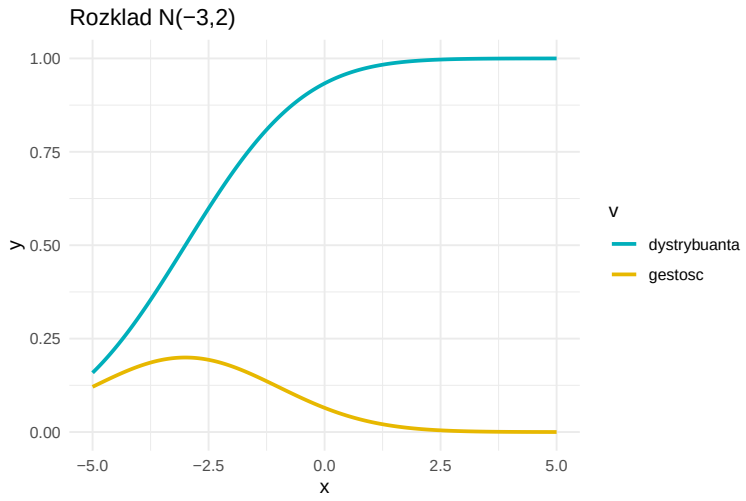
Polecenie *sample(x, n, replace=FALSE, prob=NULL)* oznacza losowe wybranie z próbki x n elementów.

1.4.3. Wykresy produkowane przez *ggplot*

```
> library(ggplot2)
> x <- seq(-5,5,by=0.01)
> y <- c(dnorm(mean=-3,sd=2,x),pnorm(mean=-3,sd=2,x))
> yname="Rozklad N(-3,2)"
> df<-data.frame(x=rep(x,2),
+                v=c(rep("gestosc",length(x)),rep("dystrybuanta",length(x))),
+                y=y)
```

nazwa rozkładu	dystrybuanta	kwantyl	gęstość	generator	parametry	pakiet
Beta	pbeta	qbeta	dbeta	rbeta	shape1, shape2	stats
Cauchy	pcauchy	qcauchy	dcauchy	rcauchy	location, scale	stats
Chi-kwadrat	pchisq	qchisq	dchisq	rchisq	df, ncp	stats
Dirchleta			ddirichlet	rdirichlet	shape	MCMCpack
F	pf	qf	df	rf	df1, df2, ncp	stats
Gamma	pgamma	qgamma	dgamma	rgamma	shape, rate	stats
Odwrotny Gamma			dinvgamma	rinvgamma	shape, scale	MCMCpack
Jednostajny	punif	qunif	dunif	runif	min, max	stats
Logistyczny	plogis	qlogis	dlogis	rlogis	location, scale	stats
Log-Normalny	plnorm	qlnorm	dlnorm	rlnorm	meanlog, sdlog	stats
t-Studenta	pt	qt	dt	rt	df, ncp	stats
Wielowymiarowy t-Studenta	pmt, pmvt	qmvt	dmt, dmvt	rmt, rmvt	mean, S, df	mnormt, mvtnorm
Normalny	pnorm	qnorm	dnorm	rnorm	mean, sd	stats
Wielowymiarowy Normalny	pmnorm, pmvnorm	qmvnorm	dmnorm, dmvnorm	rmnorm, rmvnorm	mean, varcov	mnormt, mvtnorm
Wykładniczy	pexp	qexp	dexp	rexp	rate	stats
Weibulla	pweibull	qweibull	dweibull	rweibull	shape, scale	stats
Wisharta			dwish	rwish	df, scale matrix	MCMCpack
Odwrotny Wisharta			diwish	riwish	df, scale matrix	MCMCpack
Uogólniony Pareto	pgpd	qgpd	dgpd	rgpd	loc, scale, shape	evd, evir
Uogólniony wart. ekstremalnej	pgev	qgev	dgev	rgev	loc, scale, shape	evd, evir
Gumbela, wartości ekstremalnej	pgumbel	qgumbel	dgumbel	rgumbel	loc, scale	evd
Dwumianowy	pbinom	qbinom	dbinom	rbinom	size, prob	stats
Ujemny dwumianowy	pnbinom	qnbinom	dnbinom	rnbinom	size, prob	stats
Geometryczny	pgeom	qgeom	dgeom	rgeom	prob	stats
Hipergeometryczny	phyper	qhyper	dhyper	rhyper	m,n,k	stats
Poissona	ppois	qpois	dpois	rpois	lambda	stats
Statystyka sumy rang Wilcoxona	pwilcox	qwilcox	dwilcox	rwilcox	n,m	stats
Statystyka sumy znakowanych rang Wilcoxona	psignrank	qsignrank	dsignrank	rsignrank	n	stats

```
> ggplot(df, aes(x = x, y = y)) +
+       geom_line(aes(color = v), linewidth = 1) +
+       scale_color_manual(values = c("#00AFBB", "#E7B800"))+
+       theme_minimal()+
+       ggtitle(yname)
```



Rozdział 2

Estymacja przedziałowa

2.1. Metody przedstawienia statystyki

```
> # dane niepogrupowane
> x<-c(1.1,1.7,1.1,1.3,1.2,1.2,1.2,1.8,1.5,1.5)
> # szereg punktowy
> table(x)

x
1.1 1.2 1.3 1.5 1.7 1.8
  2   3   1   2   1   1

> # szereg rozdzielczy
> w<-c(-Inf,1.3,1.6,Inf)
> w<-cut(x,w)
> table(w)

w
(-Inf,1.3]  (1.3,1.6] (1.6, Inf]
         6           2          2
```

2.2. Estymacja średniej

MODEL I - Badana cecha ma rozkład normalny $N(\mu,\sigma)$, o nieznanym μ i znanym σ. Z tablic standardowego rozkładu normalnego odczytujemy wartość $u(1-\alpha/2)$. Przedział ufności:

$$\mu \in [\overline{x} - u(1-\frac{\alpha}{2}) \cdot \frac{\sigma}{\sqrt{n}}, \overline{x} + u(1-\frac{\alpha}{2}) \cdot \frac{\sigma}{\sqrt{n}}]$$

MODEL II - Badana cecha ma rozkład normalny $N(\mu,\sigma)$, o nieznanych μ i σ. Z tablic rozkładu t-studenta z $n-1$ stopniami swobody odczytujemy wartość $t(1-\frac{\alpha}{2}, n-1)$. Przedział ufności:

$$\mu \in [\overline{x} - t(1-\frac{\alpha}{2}, n-1) \cdot \frac{s}{\sqrt{n-1}}, \overline{x} + t(1-\frac{\alpha}{2}, n-1) \cdot \frac{s}{\sqrt{n-1}}]$$

MODEL III - Badana cecha ma dowolny rozkład o nieznanej wartości oczekiwanej μ i nieznanym odchyleniu standardowym σ, zaś liczebność próby jest duża ($n \geq 100$). Z tablic standardowego rozkładu normalnego odczytujemy wartość $u(1-\alpha/2)$. Przedział ufności:

$$\mu \in [\overline{x} - u(1-\frac{\alpha}{2}) \cdot \frac{s}{\sqrt{n}}, \overline{x} + u(1-\frac{\alpha}{2}) \cdot \frac{s}{\sqrt{n}}]$$

```
>   estym_mean <- function (x, alpha, model, S=sd(x)/sqrt(length(x)-1)){
+     M <- mean(x)
```

```
+     n <- length(x)
+     if (model!=2){
+       c(LCL=(M-S*qnorm(1-alpha/2)) ,
+         UCL=(M+S*qnorm(1-alpha/2)) )
+     } else {
+       c(LCL=(M-S*qt(1-alpha/2, n-1)) ,
+         UCL=(M+S*qt(1-alpha/2, n-1)))
+     }
+   }
> estym_mean(rpois(70,2), 0.05, 2)
     LCL      UCL
1.636658 2.277628
```

2.3. Estymacja odchylenia standardowego

MODEL I - Badana cecha ma rozkład normalny $N(\mu, \sigma)$, o nieznanych parametrach μ i σ, zaś próba jest mała ($n \leq 50$). Z tablic rozkładu χ^2 odczytujemy wartości $\chi^2(1-\frac{\alpha}{2}, n-1)$ i $\chi^2(\frac{\alpha}{2}, n-1)$. Przedział ufności:

$$\sigma^2 \in [\frac{ns^2}{\chi^2(1-\frac{\alpha}{2}, n-1)}, \frac{ns^2}{\chi^2(\frac{\alpha}{2}, n-1)}]$$

MODEL II - Badana cecha ma rozkład normalny $N(\mu, \sigma)$, o nieznanych parametrach μ i σ, zaś próba jest duża ($n > 50$). Z tablic standardowego rozkładu normalnego odczytujemy wartość $u(1-\alpha/2)$. Przedział ufności:

$$\sigma^2 \in [\frac{2ns^2}{(\sqrt{2n-3} + u(1-\frac{\alpha}{2}))^2}, \frac{2ns^2}{(\sqrt{2n-3} - u(1-\frac{\alpha}{2}))^2}]$$

```
> estym_sigma <- function (x, alpha, M=mean(x)) {
+   n <- length(x)
+   S2<- var(x)
+   if (n<=50) {
+     c(LCL=sqrt(n*S2/qchisq(1-alpha/2, n-1)) ,
+       UCL=sqrt(n*S2/qchisq(alpha/2 , n-1)))
+   } else {
+     c(LCL=sqrt(2*n*S2)/(sqrt(2*n-3)+qnorm(1-alpha/2)) ,
+       UCL=sqrt(2*n*S2)/(sqrt(2*n-3)-qnorm(1-alpha/2)) )
+   }
+ }
> estym_sigma(rnorm(70,3.2,2.2), 0.05)
```

```
    LCL     UCL
1.96815 2.75986
```

2.4. Estymacja wskaźnika struktury (frakcji)

Próbka jest duża ($n \geq 100$). Z tablic standardowego rozkładu normalnego odczytujemy wartość $u(1-\alpha/2)$, oraz wyznaczamy wartość $\hat{p} = \frac{m}{n}$, gdzie m jest liczbą elementów w próbie, które posiadają badaną cechę. Przedział ufności:

$$p \in [\hat{p} - u(1-\frac{\alpha}{2})\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + u(1-\frac{\alpha}{2})\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}]$$

```
> estym_frac <- function (m, n, alpha ){
+   p<-m/n
+   if ( n>=100) {
+     c(LCL=p-qnorm(1-alpha/2)*sqrt((1-p)*p/n ),
+       UCL=p+qnorm(1-alpha/2)*sqrt((1-p)*p/n ))
+   }
+ }
> estym_frac(27,50,0.01)
```

Rozdział 3

Testy parametryczne hipotez

3.1. Testy porównania średniej z zadaną wartością

3.1.1. Test z

Założenia:

1. Badana cecha ma rozkład normalny $N(\mu, \sigma)$,
2. Parametr σ jest znany,
3. Dane na skali interwałowej lub ilorazowej.

lub

1. Badana cecha X ma dowolny rozkład
2. Próba jest duża ($n \geq 100$),
3. Dane na skali interwałowej lub ilorazowej.

Stawiamy hipotezę zerową $H_0: \mu = \mu_0$

Statystyka testowa	H_1	Obszar krytyczny W
$U = \frac{\overline{x}-\mu_0}{\sigma}\sqrt{n}$	$\mu \neq \mu_0$	$(-\infty, -u(1-\frac{\alpha}{2})] \cup [u(1-\frac{\alpha}{2}, +\infty)$
	$\mu < \mu_0$	$(-\infty, -u(1-\alpha)]$
	$\mu > \mu_0$	$[u(1-\alpha), +\infty$

```
> z.test <- function (x, sig, m0=0, alpha=0.05, alternative="two.sided") {
+   M<-mean(x)
+   n<-length(x)
+   S<-sqrt(sig/n)
+   statistic<- (M=m0)/S
+   if(alternative=="two.sided")   { 2*pnorm(abs(statistic), lower.tail = FA
+     } else if (alternative=="less") { pnorm(statistic, lower.tail=TRUE )
+     } else                          { pnorm(statistic, lower.tail=FALSE )
+     }
+ }
> z.test.power <- function (x, nsim =1000, m=m0, s=sig, alpha =0.05 ,
+                                          alternative="two.sided "){
+   n<-length(x)
+   ts<-replicate(nsim, z.test(rnorm(n,m,s ),s,m,alpha ,
+                 alternative)<alpha )
+   sum(ts)/nsim
+ }
> x<-rnorm(150,1.1,2.5)
> z.test(x, 2.5, 1.0, 0.01)

[1] 9.485738e-15

> z.test.power(x, 1000, 1.0, 2.5, 0.01)

[1] 1
```

3.1.2. Test t

Założenia:

1. Badana cecha X ma rozkład normalny $N(\mu, \sigma)$
2. Parametr σ jest nieznany,
3. Dane na skali interwałowej lub ilorazowej.

Stawiamy hipotezę zerową $H_0: \mu = \mu_0$

Statystyka testowa	H_1	Obszar krytyczny W
$t = \frac{\overline{x}-\mu_0}{s}\sqrt{n-1}$	$\mu \neq \mu_0$	$*$
	$\mu < \mu_0$	$(-\infty, -t(1-\alpha, n-1)]$
	$\mu > \mu_0$	$[t(1-\alpha, n-1), +\infty)$

$$* = (-\infty, -t(1-\frac{\alpha}{2}, n-1)] \cup [t(1-\frac{\alpha}{2}, n-1), +\infty)$$

```
> sig <-5; alpha<-0.05; m0<-2.8
> x<-rnorm(30, mean=3.2, sd=sig )
> w<-t.test(x , mu=m0, alternative="two.sided" , conf.level=1-alpha )
> str(w)

List of 10
 $ statistic  : Named num -0.234
  ..- attr(*, "names")= chr "t"
 $ parameter  : Named num 29
  ..- attr(*, "names")= chr "df"
 $ p.value    : num 0.817
 $ conf.int   : num [1:2] 0.963 4.26
  ..- attr(*, "conf.level")= num 0.95
 $ estimate   : Named num 2.61
  ..- attr(*, "names")= chr "mean of x"
 $ null.value : Named num 2.8
  ..- attr(*, "names")= chr "mean"
 $ stderr     : num 0.806
 $ alternative: chr "two.sided"
 $ method     : chr "One Sample t-test"
 $ data.name  : chr "x"
 - attr(*, "class")= chr "htest"

> c(w$statistic, w$parameter, w$p.value)

         t         df
-0.2337942 29.0000000  0.8167874
```

3.1.3. Moc testu t

```
> library(pwr)
> pwr.t.test ( n=length(x) ,
+              d=(mean(x)-m0)/sig,
+              sig.level=0.05 ,
+              power=NULL ,
+              type="one.sample", alternative="two.sided")

     One-sample t test power calculation

              n = 30
              d = 0.03769144
      sig.level = 0.05
          power = 0.05458085
    alternative = two.sided

> beta <-0.2
> pwr.t.test( n=length(x),
+             d=(mean(x)-m0)/sig ,
+                sig.level=NULL ,
+                power=1-beta ,
+                type="one.sample", alternative="two.sided")

     One-sample t test power calculation

              n = 30
              d = 0.03769144
      sig.level = 0.795811
          power = 0.8
    alternative = two.sided

> alpha<-0.1; beta<-0.1
> pwr.t.test( n=NULL ,
+             d=(mean(x)-m0)/sig ,
+                sig.level=alpha ,
+                power=1-beta ,
+                type="one.sample", alternative="two.sided")

     One-sample t test power calculation

              n = 6029.437
              d = 0.03769144
      sig.level = 0.1
```

```
        power = 0.9
  alternative = two.sided
```

3.2. Test porównania wariancji z zadaną wartością

Założenia:
1. Badana cecha X ma rozkład normalny $N(\mu, \sigma)$,
2. Próba jest mała ($n < 50$),
3. Dane na skali interwałowej lub ilorazowej.

Stawiamy hipotezę zerową $H_0: \sigma^2 = \sigma_0^2$

Statystyka testowa	H_1	Obszar krytyczny W
$\chi^2 = \frac{ns^2}{\sigma_0^2}$	$\sigma^2 \neq \sigma_0^2$	$(0, \chi^2(\frac{\alpha}{2}, n-1)] \cup [\chi^2(1-\frac{\alpha}{2}, n-1), +\infty)$
	$\sigma^2 < \sigma_0^2$	$(0, \chi^2(\alpha, n-1)]$
	$\sigma^2 > \sigma_0^2$	$[\chi^2(1-\alpha, n-1), +\infty)$

```
> alpha<-0.05; sig<-2.8; s<-3.2
> x<-rnorm(30, mean=3.2, sd=sig)
> library(EnvStats)
> w<-varTest (x, alternative = "two.sided" ,
+             conf.level = 1-alpha ,
+             sigma.squared = s^2)
> c(w$statistic, w$parameters, w$p.value)

Chi-Squared          df
  21.804084   29.000000    0.343579
```

3.3. Test porównania frakcji z zadaną wartością

Założenia:
1. Próba jest mała $n < 100$.

Stawiamy hipotezę zerową $H_0: p = p_0$

Statystyka testowa	H_1	Obszar krytyczny W
U^*	$p \neq p_0$	$(-\infty, -u(1-\frac{\alpha}{2})] \cup [u(1-\frac{\alpha}{2}), +\infty)$
	$p < p_0$	$(-\infty, -u(1-\alpha)]$
	$p > p_0$	$[u(1-\alpha), +\infty)$

gdzie

$$U^* = (2 \arcsin \sqrt{\frac{m}{n}} - 2 \arcsin \sqrt{p_0}) \sqrt{n}$$

```
> alpha <-0.05; p0<-0.6; n<-300
> x<-rnorm(300, mean=3.2, sd=5)     # ktore wieksze od 3.6
```

```
> y<-sum(x>3.6)
> w1<-prop.test (y, length(x) , p=p0 ,
+                alternative="two.sided" ,
+                conf.level=1-alpha )
> w1<-c(w1$statistic, w1$parameter, p.value=w1$p.value)
> # n p1 p2 power
> w1<-c(w1, beta=1-power.prop.test(n-length(x), p1=y/length(x), p2=p0)$power
> w1
   X-squared           df      p.value         beta
1.750347e+01 1.000000e+00 2.867835e-05 9.758275e-01
```

3.4. Test porównania dwóch średnich

Założenia:

1. Badana cecha X_1, X_2 ma w dwóch populacjach rozkłady normalne $X_1 = N(\mu_1, \sigma_1)$ i $X_2 = N(\mu_2, \sigma_2)$ (skala interwałowa lub ilorazowa),
2. σ_1 i σ_2 są znane.

Stawiamy hipotezę zerową $H_0: \mu_1 = \mu_2$

Statystyka testowa	H_1	Obszar krytyczny W
$U = \frac{\overline{x_1} - \overline{x_2}}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$	$\mu_1 \neq \mu_2$	$(-\infty, -u(1-\frac{\alpha}{2})] \cup [u(1-\frac{\alpha}{2}), +\infty)$
	$\mu_1 < \mu_2$	$(-\infty, -u(1-\alpha)]$
	$\mu_1 > \mu_2$	$[u(1-\alpha), +\infty)$

Założenia:

1. Próba jest duża ($n_1 \geq 100, n_2 \geq 100$),
2. Skala interwałowa lub ilorazowa

Stawiamy hipotezę zerową $H_0: \mu_1 = \mu_2$

Statystyka testowa	H_1	Obszar krytyczny W
$U = \frac{\overline{x_1} - \overline{x_2}}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$	$\mu_1 \neq \mu_2$	$(-\infty, -u(1-\frac{\alpha}{2})] \cup [u(1-\frac{\alpha}{2}), +\infty)$
	$\mu_1 < \mu_2$	$(-\infty, -u(1-\alpha)]$
	$\mu_1 > \mu_2$	$[u(1-\alpha), +\infty)$

Założenia:

1. Badana cecha X_1, X_2 ma w dwóch populacjach rozkłady normalne $X_1 = N(\mu_1, \sigma_1)$ i $X_2 = N(\mu_2, \sigma_2)$ (skala interwałowa lub ilorazowa),
2. σ_1 i σ_2 są nieznane, ale $\sigma_1 = \sigma_2$ (homogeniczność wariancji)

Stawiamy hipotezę zerową $H_0: \mu_1 = \mu_2$

Statystyka testowa	H_1	Obszar krytyczny W
$t = \frac{\overline{x_1} - \overline{x_2}}{\sqrt{\frac{n_1 s_1^2 + n_2 s_2^2}{n_1 + n_2 - 2}(\frac{1}{n_1} + \frac{1}{n_2})}}$	$\mu_1 \neq \mu_2$	*
	$\mu_1 < \mu_2$	$(-\infty, -t(1-\alpha, n_1+n_2-2)]$
	$\mu_1 > \mu_2$	$[t(1-\alpha, n_1+n_2-2), +\infty)$

$$* = (-\infty, -t(1-\frac{\alpha}{2}, n_1+n_2-2)] \cup [t(1-\frac{\alpha}{2}, n_1+n_2-2), +\infty)$$

```
> x<-rnorm(30,0,1)
> y<-rpois(50,1)
> t.test(x,y,paired=FALSE,var.equal=TRUE)

        Two Sample t-test

data:  x and y
t = -4.205, df = 78, p-value = 6.904e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.3694684 -0.4893923
sample estimates:
 mean of x  mean of y
0.03056962 0.96000000

> x<-rpois(70,1.2)
> t.test(x,y,paired=FALSE,var.equal=FALSE)

        Welch Two Sample t-test

data:  x and y
t = 0.75291, df = 116.5, p-value = 0.453
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.2282705  0.5082705
sample estimates:
mean of x mean of y
     1.10      0.96
```

3.5. Testy jednorodności wariancji

3.5.1. Test F Fishera

Założenia:

(i) Próbki są niezależne wybrane losowo.

(ii) Rozkład obu próbek jest normalny (WAŻNE!!!)

Wtedy

$$F = \frac{s_1^2}{s_2^2},$$

ma rozkład *FSnedecora* z $(n-1, m-1)$ stopniami swobody.

```
> x<-rnorm(30,0,5); y<-rnorm(50,1,4.8)
> # ratio to postulowana wartosc F
> var.test(x,y,ratio=1)

        F test to compare two variances

data:  x and y
F = 2.1059, num df = 29, denom df = 49, p-value = 0.02098
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 1.119310 4.191462
sample estimates:
ratio of variances
          2.105887
```

3.5.2. Test Levene'a - dwie lub więcej grup niezależnych

(i) Próbki są niezależne wybrane losowo.
(ii) W każdej grupie rozkład jest normalny $N(\mu_i, \sigma_i)$ z nieznanym μ_i (Chociaż możliwe są niewielkie odstępstwa od normalności)

Statystyka testowa

$$F = \frac{N-1}{k-1} \frac{\sum_{i=1}^{k} n_i (\hat{Z}_i - \hat{Z})^2}{\sum_{i=1}^{k} \sum_{j=1}^{n_i} (Z_{ij} - \hat{Z}_i)^2},$$

ma rozkład Fishera z $(k-1, N-k)$ stopniami swobody, gdzie $N = \sum_{i=1}^{k} n_i$, $\hat{Z}_i$ oraz $\hat{Z}$ średnie wewnątrzgrupowe i międzygrupowe statystyki $\{Z_{ij} = |X_{ij} - \hat{X}_i|, \hat{X}_i = \sum_{j=1}^{n_i} X_{ij}/n_i, 1 \leq i \leq k$. Test Levene'a został potem zmodyfikowany przez Browna i Forsythe'a (1974) poprzez zastąpienie średniej wewnątrzgrupowej $\hat{X}_i$ w definicji Z_{ij} medianą lub średnią uciętą. Autorzy modyfikacji testu Levene'a zalecają przede wszystkim stosowanie procedury z medianą grupową, gdyż zapewnia ona, ich zdaniem, test o dość wysokiej mocy, który jest odporny na brak normalności w rozkładzie danych.

```
> library(car)
> leveneTest(weight ~ group, data = PlantGrowth)

Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  2  1.1192 0.3412
      27

> leveneTest(len ~ supp*as.factor(dose), data = ToothGrowth)
```

```
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  5  1.7086 0.1484
      54
```

Parametr `location = c("median", "mean", "trim.mean")` w poleceniu `leveneTest` daje wspomnianą poprawkę Browna-Forsythe'a. Parametr `bootstrap = FALSE` wskazuje czy bootstrap ma być użyty.

3.6. Test porównania dwóch wskaźników struktur

Założenia:

1. Próba jest duża $n_1 \geq 100,\ n_2 \geq 100$.
   Stawiamy hipotezę zerową $H_0 :\ p_1 = p_2$

Statystyka testowa	H_1	Obszar krytyczny W
$U = \frac{p_1^* - p_2^*}{\sqrt{\frac{p^*(1-p^*)}{n^*}}}$	$p_1 \neq p_2$	$(-\infty, -u(1-\frac{\alpha}{2})] \cup [u(1-\frac{\alpha}{2}), +\infty)$
	$p_1 < p_2$	$(-\infty, -u(1-\alpha)]$
	$p_1 > p_2$	$[u(1-\alpha), +\infty)$

gdzie

$$p_1^* = \frac{m_1}{n_1},\ \ p_2^* = \frac{m_2}{n_2},\ \ p^* = \frac{m_1 + m_2}{n_1 + n_2},\ \ n^* = \frac{n_1 n_2}{n_1 + n_2}$$

Założenia:

1. Próba jest mała $n_1 < 100$, lub $n_2 < 100$.
   Stawiamy hipotezę zerową $H_0 :\ p_1 = p_2$

Statystyka testowa	H_1	Obszar krytyczny W
U^*	$p_1 \neq p_2$	$(-\infty, -u(1-\frac{\alpha}{2})] \cup [u(1-\frac{\alpha}{2}), +\infty)$
	$p_1 < p_2$	$(-\infty, -u(1-\alpha)]$
	$p_1 > p_2$	$[u(1-\alpha), +\infty)$

gdzie

$$U^* = (2 \arcsin \sqrt{p_1^*} - 2 \arcsin \sqrt{p_2^*}) \sqrt{n^*},$$

oraz

$$p_1^* = \frac{m_1}{n_1},\ \ p_2^* = \frac{m_2}{n_2},\ \ n^* = \frac{n_1 n_2}{n_1 + n_2}$$

```
> palacy  <- c( 83, 120 )
> pacjenci <- c( 96, 128 )
> prop.test(palacy, pacjenci)
```

```
        2-sample test for equality of proportions with continuity correction

data:  palacy out of pacjenci
X-squared = 2.6284, df = 1, p-value = 0.105
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.16230216  0.01646883
sample estimates:
   prop 1    prop 2
0.8645833 0.9375000
```

Rozdział 4

Inne testy

4.1. Test badania normalności rozkładu

```
> x<-rnorm(150,0.5,3)
> shapiro.test(x)

        Shapiro-Wilk normality test

data:  x
W = 0.99617, p-value = 0.9668

> ks.test(scale(x),"pnorm")

        Asymptotic one-sample Kolmogorov-Smirnov test

data:  scale(x)
D = 0.036724, p-value = 0.9875
alternative hypothesis: two-sided

> library(nortest)
> lillie.test(x)

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  x
D = 0.036724, p-value = 0.8902

> library(fBasics)
> library(interp)
> jbTest(x)

Title:
 Jarque - Bera Normality Test

Test Results:
  PARAMETER:
    Sample Size: 150
  STATISTIC:
    LM: 0.044
    ALM: 0.016
  P VALUE:
    LM p-value: 0.977
    ALM p-value: 0.992
    Asymptotic: 0.978
```

4.2. ANOVA

W tej części opracowania zostanie przedstawiony zostanie sposób porównania więcej niż dwóch zmiennych niezależnych. Gdy zmienne w grupach mają rozkład normalny oraz równe wariancje w grupach można stosować klasyczny test ANOVA. Hipoteza $H_o: \ \mu_1 = \mu_2 = \ldots = \mu_k$ wobec hipotezy przeciwnej $\exists_{i \neq j} \mu_i \neq \mu_j$.

```
> set.seed(4489)
> y=c(rnorm(14,0,3),rnorm(12,5,3),rnorm(15,9,3),rnorm(11,11,3))
> g=factor(rep(LETTERS[1:4],c(14,12,15,11)))
> # normalno\'s\'c rozk\{}lad\'ow
> tapply(y,g,function(x) shapiro.test(x)$p.value)

        A         B         C         D
0.7523428 0.8314125 0.7737079 0.4740450

> # jednorodno\'s\'c wariancji
> leveneTest(y,g,location ="trim.mean",trim.alpha=0.2, bootstrap=T, num.boot

NULL

> summary(aov(y~g))

            Df Sum Sq Mean Sq F value   Pr(>F)
g            3 1009.8   336.6   29.35 6.32e-11 ***
Residuals   48  550.5    11.5
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Wszystkie testy potwierdzają normalność próbek i jednorodność wariancji ale test ANOVA wyklucza hipotezę H_o.