```
ip <- installed.packages()[, 1] #   показва инсталираните пакети
pfi <- setdiff(c("ggplot2", "ggpubr", "nortest"), ip)
#   Форният ред показва кои пакети не са инсталирани

if(length(pfi) > 0) {
  install.packages(pfi)
}

library(ggplot2)
library(ggpubr)
library(nortest)  #   Тестове за проверка на нормално разпределение



#           Статистически заключения
#       Статистическите заклюения, основаващи се на случайни извадки, позволяват
значително
#   да се намалят разходите за статистически изследвания на големи по обем
съвкупности.
#   Информацията, получена от извадките почти винаги удовлетворява потребностите
на
#   проучващите.
#       Статистическите заключения имат две основни направления:
#           - статистическо оценяване
#           - проверка на хипотези
#       И двете направления имат вероятностен характер и са свързани помежду си.

#       1. Статистическо оценяване
#           1.1. Точкови оценки
#       Точковата оценка представлява отделна величнина, получена от данните на
случайна извадка,
#   която може да се доближава в различна степен до съответния параметър на
популацията
#   (генералната съвкупност). Примери за точкови оценки са  оценката на
локацията и вариацията.

set.seed(950411)
x <- rnorm(n = 200, 10, 10)
x_sample <- sample(x, 30)
mean(x)   #   10.357
mean(x_sample)  #   9.1


set.seed(620331)
y <- rbinom(n = 400, size = 1, prob = 0.3)
y_sample <- sample(y, 50)
mean(y)   #   0.3175
mean(y_sample)  #   0.34



#           1.2. Интервални оценки
#       Всяка оценка, получена от случайна изваврдка е обременена със случайна
грешка. Основният
#   недостатък на точковата оценка се състои в това, че не позволява да се
формират изводи за
#   размера на тази грешка, за точността на нейното изчисляване по отношение по
отношение на
#   обема  на вариацията на разпределението й. Тази информация се съдържа в
интервалната оценка.
#

alpha <- 0.05   #   ниво на съгласие
k <- qnorm(1 - alpha/2) #   квантил
```

```
n <- length(x_sample) # размер на извадката
mean(x_sample) + k*c(-1, 1)*sd(x_sample)/sqrt(n)  #   доверителен интервал

n <- length(y_sample) # размер на извадката
mean(y_sample) + k*c(-1, 1)*sd(y_sample)/sqrt(n)  #   доверителен интервал


#           1.3. Обем  на извадката
#       Обемът на извадката е един от най-важните фактори за точността на тези
оценки


mu <- mean(x)
d <- density(x)
ss <- c(34, 7, 21)
counter <- 0
par(mfrow = c(2, 2))
for(i in c(10, 30, 100)) {
  counter <- counter + 1
  plot(d, main = paste("Density plot - ", i, "obs"), xlab = "x", lwd = 2)
  set.seed(ss[counter])
  xx <- sample(x, i)
  abline(v = mu)
  abline(v = mean(xx) + k*c(-1, 0, 1)*sd(xx)/sqrt(i), col = "red", lwd = 1.5,
lty = 2)
}
par(mfrow = c(1, 1))










#         2. Тестване на хипотези

#         2.1. Хипотези. Видове  хипотези

#       При статистическите хипотези се проверява правдоподобността на
предварително
#   формулирани предположения относно праметрите или вида на неизвестното
разпределение
#   в популцията. Заключенията, основаващи се на хипотезите имат вероятностен
храктер.
#       Проверката на хипотеза се извършва в няколко стъпки. Стартира (първа
стъпка) с
#   формулирането на две хипотези - нулева (H0) и алтернативна (H1). Двете
хипотези са
#   взаимоизключващи се.
#       Съществуват три вида хипотези:
#           - Двустранна: H0: параметър = C / H1: параметър != C
#           - Едностранна (лявостранна): H0: параметър >= C / H1: параметър < C
#           - Едностранна (дясностранна): H0: параметър <= C / H1: параметър > C

#       Втората стъпка е да се избере нивото на съгласие (alpha). Това е
вероятност, която
#   определя зоната за отхвърляне на нулевата хипотеза. Alpha се определя
предварително
#   в съответствие с целите и здачите на изследването. Най-често нивото н
```

```
съгласие е 0.05.
x <- seq(-4, 4, by = 0.01)
d <- dnorm(x)
alpha <- 0.05

rej <- paste0("Отхвърлям (alpha = ", alpha, ")")
criteria <- factor(rep("Не отхвърлям", length(x)), levels = c("Не отхвърлям",
rej))
criteria[which(x < qnorm(alpha))] <- rej
hypothesis_greater <- qplot(x, d, geom = c("path", "area"), fill = criteria,
xlab = "Z",
                           ylab = "Плътност", main = "H0:    Параметър > C") +
  scale_fill_manual(values = c("darkgreen", "red"))
hypothesis_greater



criteria <- factor(rep("Не отхвърлям", length(x)), levels = c("Не отхвърлям",
rej))
criteria[which(x > qnorm(1 - alpha))] <- rej
hypothesis_less <- qplot(x, d, geom = c("path", "area"), fill = criteria, xlab =
"Z",
                         ylab = "Плътност", main = "H0:    Параметър < C") +
  scale_fill_manual(values = c("darkgreen", "red"))
hypothesis_less



rej1 <- paste0("Отхвърлям (<) (alpha = ", alpha/2, ")")
rej2 <- paste0("Отхвърлям (>) (alpha = ", alpha/2, ")")
criteria <- factor(rep("Не отхвърлям", length(x)), levels = c("Не отхвърлям",
rej1, rej2))
criteria[which(x > qnorm(1 - alpha/2))] <- rej2 #| x < qnorm(0.025))] <-
"Отхвърлям"
criteria[which(x < qnorm(alpha/2))] <- rej1 #
hypothesis_two_sided <- qplot(x, d, geom = c("path", "area"), fill = criteria,
xlab = "Z",
                              ylab = "Плътност", main = "H0:    Параметър = C")
+
  scale_fill_manual(values = c("darkgreen", "red", "darkred"))
hypothesis_two_sided


#   В червено са изобразени критичните области, при които нулевата хипотеза се
отхвърля
#   Нивото на съглсие alpha = 0.05

#       Трета и четвърта стъпка са да се определят се емперичната характеристика
и след това
#   да се провери дали попада в критичната област. По-лесният вариант е да се
види стойността
#   на p-value (significance).
#       Какво по-точно представлява p-value?
#       Най-грубо казано, с подхода p-value първо оценяваме колко вероятно е
емпиричнаата
#   стойност, получена от статистическия тест при положение, че нулевата
хипотеза е вярна.
#   Критерият за взимане на решение дали да се отхвърли H0 включва сравнение
натзи вероятност
#   с определеното ниво на съгласие alpha.


#         2.2. Тестове за локция/очакване на разпределението
```

```
#   Пример
#     Георги (наскоро формиран баровец) казал на Гергана, че средното
разстояние, което изминава
#   топката за голф при негов удар е 247 метра. Естестено, Гергана (учила през
живота си поне
#   един курс по статистика) е скептична и му иска доказателство. Така не Георги
му се наложило
#   да направи 25 опита, които той стриктно си записал във вектора
golf_driving_distances <- c(239, 229, 223, 224, 267, 235, 264, 235, 239, 251,
200,
                            191, 254, 253, 238, 216, 256, 228, 247, 219, 245,
251, 235, 246, 266)


#   Както не веднъж сме споменавали, оценките на статистиките биват параметрични
и непараметрични,
# в зависимост от вида на разпределението, за което ги изчисляваме.
Параметричната статистика се
# изпозлва при наличие на НОРМАЛНО разпределение или поне симетрично
разпределение, за което нямаме
# голям брой екстремални стойности. Ето защо, първата задача е да изследваме
вида на разпределението

#   Как можем да проверим едно разпределение дали е нормално или не?
par(mfrow = c(2, 2))
  qqnorm(golf_driving_distances); qqline(golf_driving_distances)

  d <- density(golf_driving_distances)
  hist(golf_driving_distances, main = "Хистограма", col = "red", xlab = "Golf
driving distances",
       prob = T, ylim = c(0, max(d$y)))
  lines(d, lw = 2)
  x_axis <- seq(0.9*min(golf_driving_distances),
1.11*max(golf_driving_distances), length = 300)
  y_axis <- dnorm(x_axis, mean = mean(golf_driving_distances), sd =
sd(golf_driving_distances))
  lines(x_axis, y_axis, col = "blue", lw = 2)

  boxplot(golf_driving_distances, horizontal = TRUE)
par(mfrow = c(1, 1))

ggqqplot(golf_driving_distances)  # Друг начин за Q-Q plot


shapiro.test(golf_driving_distances)
# Нулевата хипотеза на теста (H0) е, че разпределението е нормално
# Стойността на p-value = 0.4157 => не можем да отхвърлим H0 =>
# приемаме, че разпределението е нормално

gdd <- golf_driving_distances
y <- rnorm(n = length(gdd), mean = mean(gdd), sd = sd(gdd))
ks.test(x = golf_driving_distances, y = y)

ks.test(x = scale(golf_driving_distances), y = "pnorm")

ad.test(x = golf_driving_distances)

#       Тестовете и графиките показват, че разпределението е нормално.
Следователно най-добре е
#   да използваме параметрични тестове, т.е. student t тест



#       2.1.1. Параметрични тестове за една извадка
```

```
#   H0: mean(x) = 247
#   H1: mean(x) != 247

#   Определяме ниво на съгласие alpha = 0.05

# x - приема вектор (задължителен параметър)
# y - приема вектор (не е задължителен)
# alternative - отговаря за типа на хипотезата и приема стойностите
c("two.sided", "less", "greater")
# mu - константа, с която искаме да тестваме нулевата хипотеза

t.test(x = golf_driving_distances, mu = 247, alternative = "two.sided")
#       Стойността на p-value < alpha => отхвърляме нулевата хипотеза H0. Тоест
средната
#   не е равна на 247 метра.
#       Всички t тестове ни показват и доверителните интервали на очакването.
Ако проверяваната
#   стойност (mu) е извън този доверителен интервал, то отхвърляме H0 в полза на
H1.

#       Можем да порменяме големината на доверителните интервали с помощта на
параметъра
#   conf.level, където посочваме с каква вероятност искаме да присъства
очакването в него
t.test(x = golf_driving_distances, mu = 247, alternative = "two.sided",
       conf.level = 0.9)



#       Пример 2
#       Службата за вътрешни приходи (IRS) публикува данни за федералните
данъчни декларации за
#   доходите на физическите лица. Извадка от 12 лица от последната година показа
коригираните
#   брутни доходи в хиляди долари, които са записани във вектора
incomes <- c(9.7, 93.1, 33.0, 21.2, 81.4, 51.1, 43.5, 10.6, 12.8, 7.8, 18.1,
12.7)

#       Искаме да проверим дали физическите лица получават годишно поне 20 000
долара?

qqnorm(incomes);  qqline(incomes)
#   От Q-Q plot-а се вижда, че данните не са нормално разпределени. Ето защо ще
използваме
#   непараметрични тестове



#       2.1.2. Непараметрични тестове за една извадка
#   Непараметричният еквивалент на Student t тест е Wilcoxon signed rank test

#   H0: E[x] = 20
#   H1: E[x] > 20
wilcox.test(x = incomes, alternative = "greater", conf.int = TRUE, mu = 20)
#       Стойността на p-value е 0.19 > alpha = 0.05 => не можем да отхвърлим H0.
Доверителният
#   интервал съдържа стойността 20 (14.35, Inf)

#   Параметрите в Wilcoxon теста са сходни с тези на Student t тест.
Единствената разлика е
#   параметърът conf.int, който отговаря за показването на доверителния
интервал.
```

```
#       Пример 3
#       Американската асоциация на университетските преподаватели (AAUP)
провежда проучвания
#   за заплатите на професори от колежи и публикува резултатите си в годишния
доклад на AAUP
#   за икономическото състояние на професията. Да предположим, че искаме да
решат дали
#   средните заплати на преподавателите в частни и публични институции са
различни. Резултатите
#   са представени във векторите по-долу
private_institutions <- c(87.3, 75.9, 108.8, 83.9, 56.6, 99.2, 54.9, 73.1, 90.6,
89.3, 84.9,
                          84.4, 129.3, 98.8, 148.1, 132.4, 75.0, 98.2, 106.3,
131.5, 41.4,
                          115.6, 60.6, 64.6, 59.9, 105.4, 74.6, 82.0, 87.2,
45.1, 116.6,
                          106.7, 66.0, 99.6, 53.0)

public_institutions <- c(49.9, 105.7, 116.1, 40.3, 123.1, 79.3, 72.5, 57.1,
50.7, 69.9, 40.1,
                         71.7, 73.9, 92.5, 99.9, 95.1, 57.9, 97.5, 44.9, 31.5,
49.5, 55.9,
                         66.9, 56.9, 75.9, 103.9, 60.3, 80.1, 89.7, 86.7)


#       Първо ще започнем с изследването дали разпределенията са нормално
разпределени. Ако и
#   при два вектора имаме нормални разпределения, то ще изпозлваме параметрична
статистика.
#   Но, ако поне за единия вектор разпределенеито не е нормално, тогава е по-
удачно да се спрем
#   на непараметрични тестове.
shapiro.test(private_institutions)
shapiro.test(public_institutions)
#   Минималната стойност на p-value за двата вектора е 0.6798 > alpha = 0.05 =>
разпределенията и
#   на двата вектора ги приемаме за нормални.


#       2.1.3. Параметрични тестове за две извадки - Indipendent Two Sample t
test и
#   Welch Two sample t test

#   Имаме два параметрични теста за проверка на локацията на две изваадки.
Разликата между двата
#   теста е предположението, че вариациите на двете извадки са с равни вариации
(Independent Two
#   Sample t test) или че не са - Welch Two Sample t test.
#   Independent Two Sample t test е по-точен от Welch Two Sample t test

#   Тест за сравняване на вариациите на две извадки от нормално разпределена
популация
var.test(x = private_institutions, y = public_institutions)
#   Нулевата хипотеза H0 е, че двете извадки имат равна вариация.
#   В нашия случай, стойността на p-value = 0.6253 и следователно


#   H0: mean(x) - mean(y) = 0
#   H1: mean(x) - mean(y) != 0


t.test(x = private_institutions, y = public_institutions, var.equal = TRUE)
```

```
t.test(x = private_institutions, y = public_institutions)
#       И двата теста отхвърлят нулевата хипотеза, че имаме равенство между
средните стойности на
#   двете извадки (p-value = 0.0196 и p-value = 0.0188). Тоест съществува
статистически значима
#   разлика между годишните заплащания на професорите в частните и публичните
колежи. Разликата е
#   в полза на частните колежи.

#   Доверителният интервал е построен върху разликата от средните стойности на
двете изваврдки
#   и претеглена сума на вариациите.




#       Пример 4
data("mtcars")
#       Искаме да изследваме дали средната мощност на колата, измерена в конски
сили hp се
#   различава за различните трансмисии. Данните са взети от "mtcars".

nortest::ad.test(mtcars$hp[which(mtcars$am == 0)])
nortest::ad.test(mtcars$hp[which(mtcars$am == 1)])
#   Тестът за нормалност на разпределението отхвърля H0 при ръчните скорости (p-
value = 0.00149).
#   Следователно ще използваме теста на Wilcoxon за две извадки


#       2.1.4. Непараметрични тестове за две извадки

#   H0: E(x) - E(y) = 0
#   H1: E(x) - E(y) != 0


wilcox.test(mpg ~ am, data = mtcars, conf.int = TRUE, exact = FALSE)
#       Стойността на p-value за теста е 0.001871 < 0.05 = alpha => Отхвърляме
H0. Тоест
#   Съществува статистически значима разлика между очакваните мощности при
колите с ръчна и
#   автоматична трансмисии. По-мощни са колите с ръчна трансмисия.

#   Доверителният интервал е построен по-много интересна формула, която няма да
я обясняваме,
#   но я има :). Достатъчно е да знаем, че разликата (mu = 0) не попада в
интервала.




install.packages("gplots")
library(gplots)



#           Изследване на локациите на разпределеняита при повече от две
#       групи



#       Пример
#       Взета е извадка от месечни наеми на апартаметни в различни региони
#   в САЩ (в долари)

Northeast <- c(1005, 898, 948, 1181, 1244)
```

```
Midwest <- c(870, 748, 699, 814, 721, 606)
South <- c(891, 630, 861, 1036)
West <- c(1025, 1012, 1090, 926, 1269)

#       Искаме да изследваме дали между някой от регионите съществува значима
#   разлика в очакването за цените в наемите.

#     Данните трябва да ги обединим в един data frame
rent_data <- data.frame(rent = c(Northeast, Midwest, South, West),
                        region = c(rep("Northeast", length(Northeast)),
                                   rep("Midwest", length(Midwest)),
                                   rep("South", length(South)),
                                   rep("West", length(West))))




#       В предишното упражнение използвахме Student t тест и Wilcoxon тест,
#   за да изследваме средните стойности и медианите на една извадката или
#   между две групи от наблюдения.
#       За изследването на разлика между локациите на повече от две групи
#   трябва да използваме One-way ANOVA (параметричен тест) или Kruskal
#   тест (непараметричния еквавалент на ANOVA).
#       Нека имаме n на брой вектора X1, X2, ..., Xn. Тогава имаме
#   нулевата хипотеза H0: E[X1] = E[X2] = ... = E[Xn] и алтернатива
#   H1: поне при една от двойките E[Xi] != E[Xj] за i != j.

#       Като всеки един параметричне тест и One-way ANOVA има своите
#   първоначални предположения, които, ако бъдат нарушени, то трябва
#   да използваме Kruskal тест

#       Предположения
#   1. За всяка една група, разпределението на стойностите трябва да
#   бъде нормално разпределена
#   2. Статистически еднаква дисперсия при всички групи (хомогенност на
дисперсиите).



#     Ще започнем с изследване на разпределението на данните по различните
#   групи. Най-лесно проверката ще стане с помощта на функцията aggregate.
#   Като фунцкия за агрегация ще използваме теста на Shapiro-wilk
aggregate(rent ~ region, data = rent_data, FUN = function(x) {shapiro.test(x)
$p.value})

#   Минималната стойност p-value за четирите групи е 0.456 > 0.05 = alpha =>
#   не можем да отхвърлим H0 => приемаме, че и четирите групи са нормално
разпределени

#     Хомогенността на дисперсиите ще проверим с помощта на теста на Barlett,
#   с нулева хипотеза за равемство на дисперсиите между различните групи
bartlett.test(rent ~ region, data = rent_data)
#   P-value = 0.6957 > 0.05 = alpha => имаме статистически равни дисперсии

#     One-way ANOVA
summary(rent_anova <- aov(formula = rent ~ region, data = rent_data))
#   С помощта на фунцкията "aov" прилагаме One-way ANOVA. Функцията съдържа
#   параметрите formula и data.
#   Стойността на p-value = 0.0023 < 0.05 = alpha => отхвърляме H0 в полза на H1
=>
#   съществува статистически значима разлика поне в някоя от двойките.

#   Остана да видим къде между кои групи са разликите. Това лесно става графично
#   с помощта на функцията plotmeans()
```

```
plotmeans(formula = rent ~ region, data = rent_data)


#   Друга опция е използването на така наречените Post-hoc pairwise контрасти,
#   които изследват взаимодействието на една група спрямо останалите.
#   Съществуват различни методи за изследването им, но ние ще се спрем само на
#   Tukey HSD. Върнатият резултат представлява тества на разликите между
#   всички възможни две групи, където нулевата хипотеза е, че двете локации са
#   статистически равни (или, че разликата им е = 0)
(tukey <- TukeyHSD(rent_anova))
#   Съществените разлики при групите се забелязват в последната колона
#   "p adj" (p-value), където искаме стойността на p-value < alpha - нивото на
съгласие

#   Тоест групите между, които имаме разлика са (Northest, Midwest) и (West,
Midwest)

plot(tukey) #   Графично представяне на разликите между отделните групи


#   Други методи за анализ на Post hoc pairwise са
pairwise.t.test(rent_data$rent, rent_data$region, p.adj = "bonf")
pairwise.t.test(rent_data$rent, rent_data$region, p.adj = "holm")
#       ! Различните тестове, дават различни резултати при анализа. Ето защо е
важно
#   да се избере най-подходящия алгоритъм за конкретната задача.
#       Горните два теста връщат директно стойността на p-value


#     Пример
#     изследване на връзката между месец в годината и средните стойности на
озона
#   за Ню Йорк.

data(airquality)

#   Изследване за нормално разпределние в различните групи.
aggregate(Ozone ~ Month, data = airquality, FUN = function(x)
{round(shapiro.test(x)$p.value, 3)})
#   Имаме нарушение на условието за нормално разпределение на стойностите (Май и
Септември)



kruskal.test(Ozone ~ Month, data = airquality)
#   Стойността на p-value за теста е 6.901e-06 << alpha = 0.05 => съществува
#   статистически значима разлика между групите.

#     Post-hoc анализ за Kruskal-Wallis тест
pairwise.wilcox.test(airquality$Ozone, airquality$Month,
                     p.adjust.method = "BH", exact = FALSE)
#   В получената табличка са записани стойностите на p-value при изследването на
#   разликите между групите. Така статистически значима разлика получаваме при
месеците
#   (5, 7), (5, 8), (6, 7) и т.н.
```