ОПИСАНИЕ ДАННЫХ

Для анализа был выбран набор данных "Personal Key Indicators of Heart Disease" https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease

По данным CDC, болезни сердца являются одной из ведущих причин смерти жителей США. Этот набор данных был получен от CDC в 2020 году и был составлен на основе телефонных опросов 319.795 граждан.

Описание столбцов с данными:

• HeartDisease [Yes/No] — имеет ли человек болезни сердца
• BMI — индекс массы тела (18.5-25 — норма, 16 и менее — выраженный дефицит массы тела, 40 и более — ожирение третьей степени)
• Smoking [Yes/No] — "Выкурили ли вы хотя бы 100 сигарет за всю свою жизнь?"
• AlcoholDrinking [Yes/No] — "Выпиваете ли вы более 14 порций алкоголя (мужчины) или более 7 (женщины) в неделю?"
• PhysicalHealth — "В течение скольких дней в течение последних 30 дней ваше физическое здоровье было не в порядке?"
• MentalHealth — "В течение скольких дней в течение последних 30 дней ваше психическое здоровье было не в порядке?"
• DiffWalking [Yes/No] — "Испытываете ли вы серьезные трудности при ходьбе или подъеме по лестнице?"
• Sex [Male/Female] — пол респондента
• AgeCategory — возрастная категория респондента
• PhysicalActivity [Yes/No] – "Занимались ли вы какими-либо видами спорта (бег, езда на велосипеде и т.д.) в прошлом месяце?"
• SleepTime — "Сколько часов в среднем вы спите?"

Начальная подготовка данных:

```
install.packages("readr")
library("readr")
setwd("/Users/alexandernizov/Desktop/Prac")
disease_data <- read_csv("heart_2020_cleaned.csv")
#Ограничим объём данных 2500 респондентами и удалим несколько столбцов
df <- disease_data[-c(2501:319795),-c(5,11,12,14,16,17,18)]
```

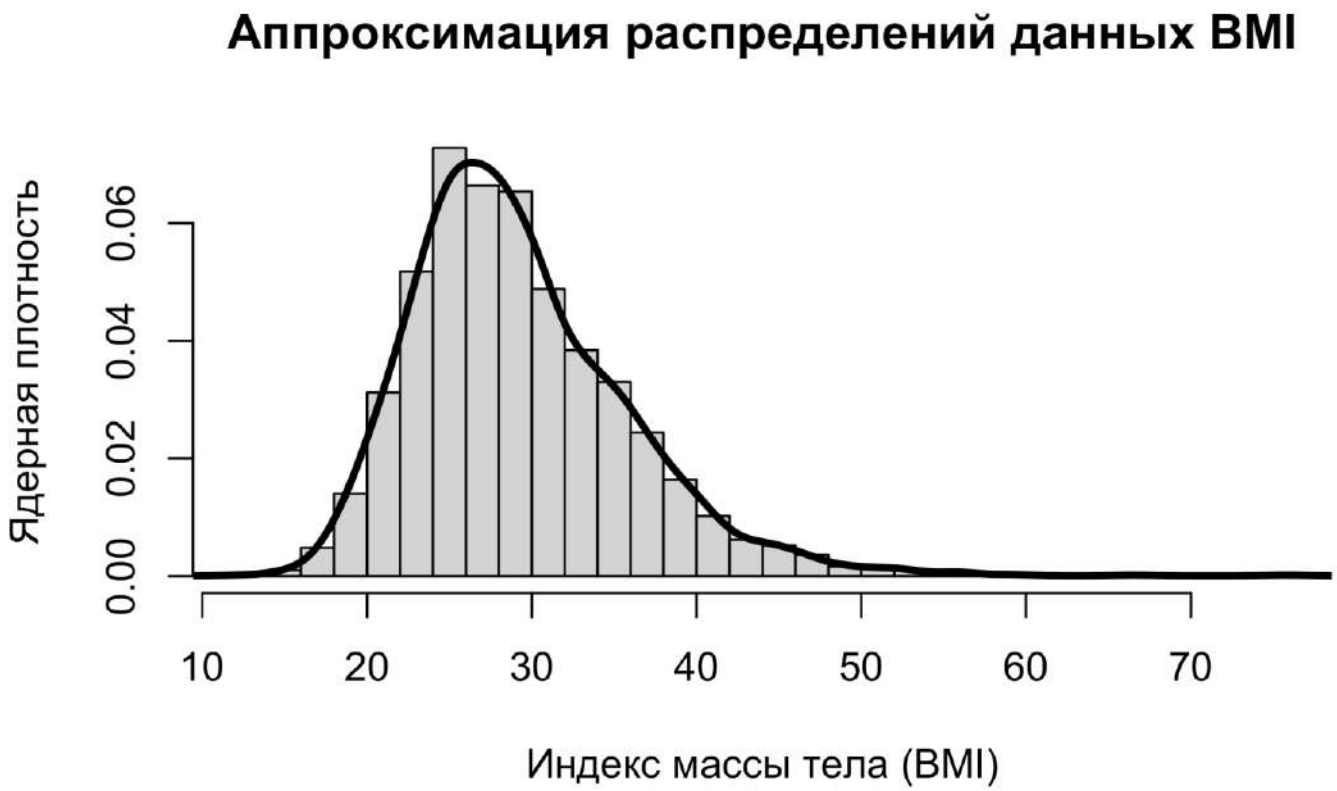
Как выглядят анализируемые данные (первые 10 строк):

	HeartDisease	BMI	Smoking	AlcoholDrinking	PhysicalHealth	MentalHealth	DiffWalking	Sex	AgeCategory	PhysicalActivity	SleepTime
1	No	16.60	Yes	No	3	30	No	Female	55-59	Yes	5
2	No	20.34	No	No	0	0	No	Female	80 or older	Yes	7
3	No	26.58	Yes	No	20	30	No	Male	65-69	Yes	8
4	No	24.21	No	No	0	0	No	Female	75-79	No	6
5	No	23.71	No	No	28	0	Yes	Female	40-44	Yes	8
6	Yes	28.87	Yes	No	6	0	Yes	Female	75-79	No	12
7	No	21.63	No	No	15	0	No	Female	70-74	Yes	4
8	No	31.64	Yes	No	5	0	Yes	Female	80 or older	No	9
9	No	26.45	No	No	0	0	No	Female	80 or older	No	5
10	No	40.69	No	No	0	0	Yes	Male	65-69	Yes	10

Задание 1. Реализовать аппроксимацию распределений данных с помощью ядерных оценок.

```r
#1) Реализовать аппроксимацию распределений данных с помощью ядерных оценок.
data_BMI <- df$BMI
hist(data_BMI, breaks = 25, freq = FALSE,
     xlab = "Индекс массы тела (BMI)", ylab = "Ядерная плотность",
     main = "Аппроксимация распределений данных BMI")
lines(density(data_BMI), lwd = 3)
```

График:



Вывод:

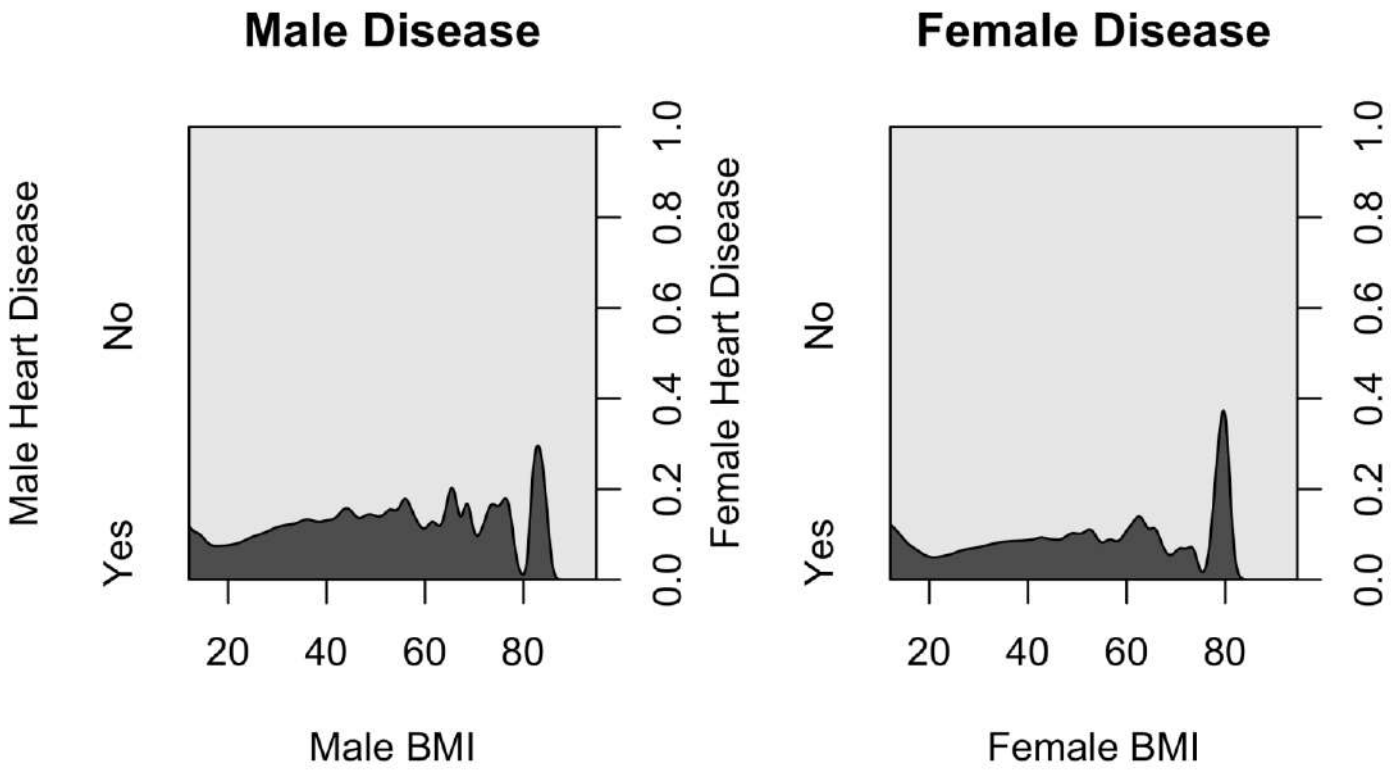
Согласно визуальной оценке графика, индекс массы тела BMI не распределён нормально.

Задание 2. Реализовать анализ данных с помощью cdplot, dotchart, boxplot и stripchart.

Cdplot:

```
#cdplot
disease_data$HeartDisease <- as.factor(disease_data$HeartDisease)
layout(matrix(1:2, ncol = 2))
cdplot(disease_data$BMI[disease_data$Sex == "Male"],
       disease_data$HeartDisease[disease_data$Sex == "Male"],
       xlab = "Male BMI",
       ylab ="Male Heart Disease",
       main ="Disease probability",
       bw=1.1)
cdplot(disease_data$BMI[disease_data$Sex == "Female"],
       disease_data$HeartDisease[disease_data$Sex == "Female"],
       xlab = "Female BMI",
       ylab ="Female Heart Disease",
       main ="Disease probability",
       bw=1.1)
```

График cdplot:



Вывод по cdplot:

При повышении индекса массы тела BMI вероятность возникновения сердечных заболеваний повышается (более выраженно у мужчин).

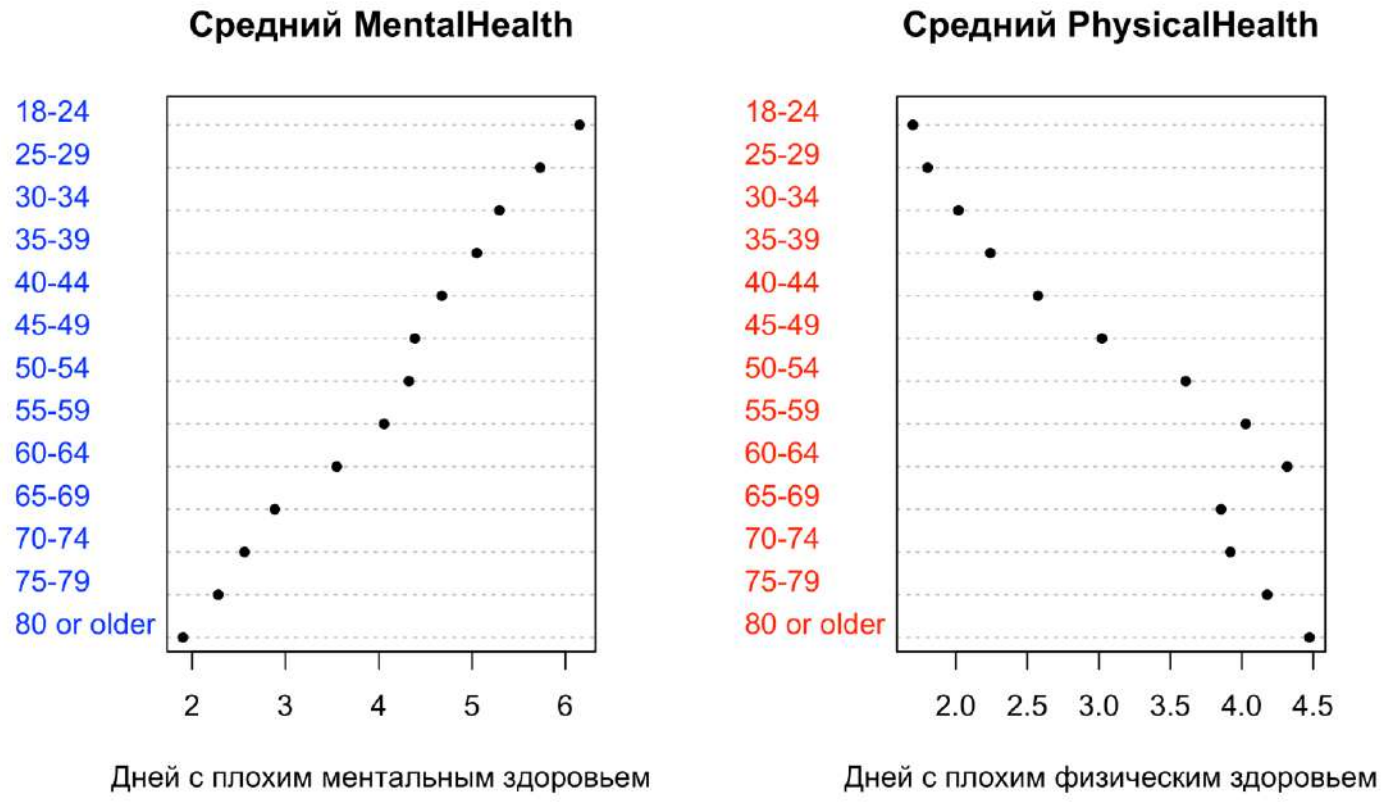
dotchart:

Dotchart

Рассмотрим две новые таблицы, в которых будут отображены средние показатели физического и ментального здоровья респондентов по всем возрастным категориям. Визуализируем показатели.

```
#dotchart
y <- factor(disease_data$AgeCategory)
newdate_mental <- aggregate(disease_data$MentalHealth,
                            by = list(y), mean)
newdate_physical <- aggregate(disease_data$PhysicalHealth,
                              by = list(y), mean)
colnames(newdate_mental) <- c('AgeCategory', 'MeanMentalHealth')
colnames(newdate_physical) <- c('AgeCategory', 'MeanPhysicalHealth')

layout(matrix(1:2, ncol = 2))
dotchart(newdate_mental$MeanMentalHealth,
         groups = newdate_mental$AgeCategory, gcolor = "blue", pch = 20,
         main="Средний MentalHealth",
         xlab="BMI", cex = 1)
dotchart(newdate_physical$MeanPhysicalHealth,
         groups = newdate_physical$AgeCategory, gcolor = "red", pch = 20,
         main="Средний PhysicalHealth",
         xlab="BMI", cex = 1)
```

График dotchart:

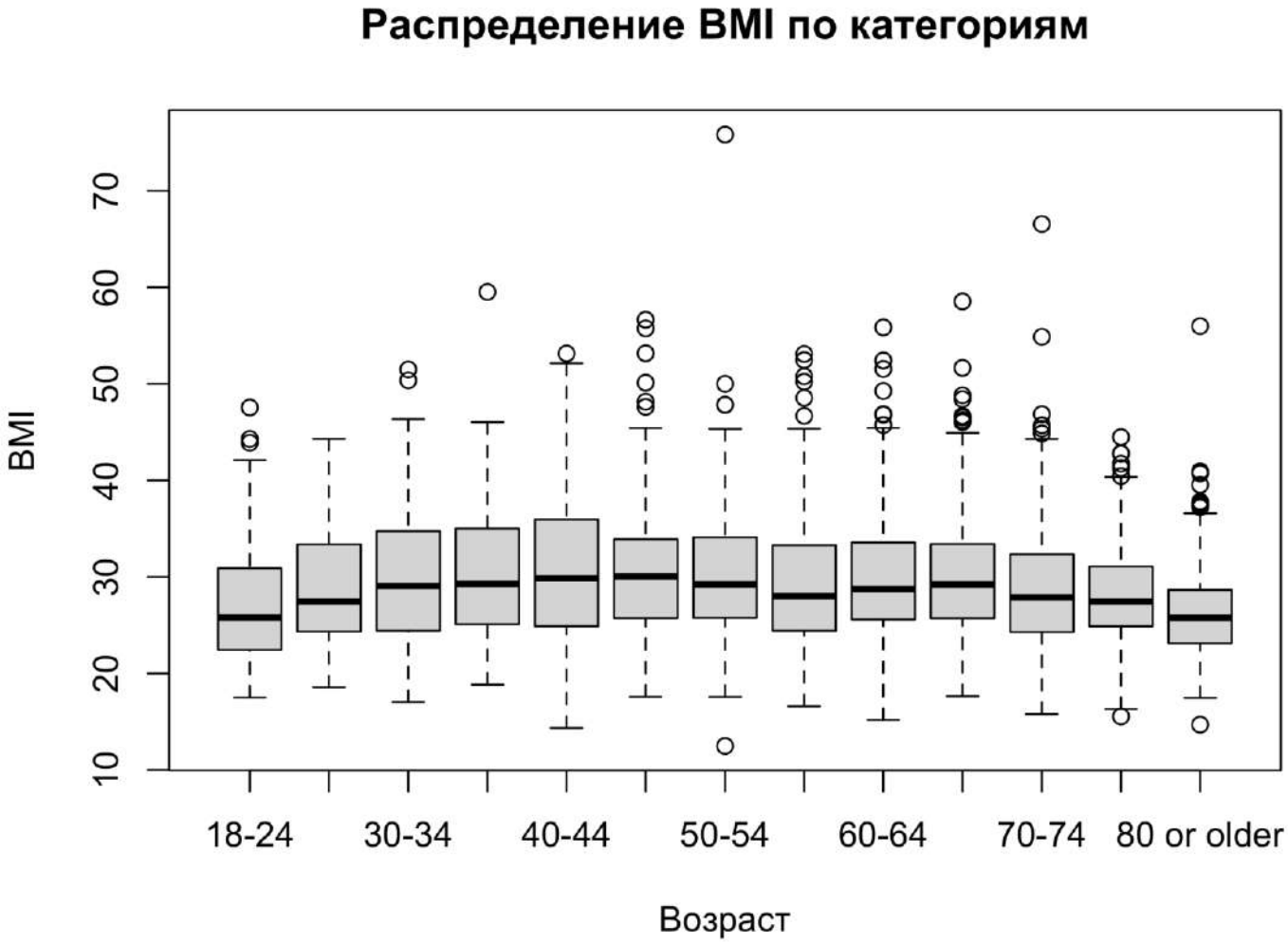


Вывод по dotchart:
Более молодые респонденты в среднем хуже отзываются о своём ментальном здоровье. В случае с физическим здоровьем всё (ожидаемо) наоборот — чем старше человек, тем больше проблем.

Boxplot

```
#boxplot
boxplot(BMI ~ AgeCategory, xlab = "Возраст", ylab = "BMI",
        main = "Распределение BMI по категориям", data = df)
```

График boxplot:



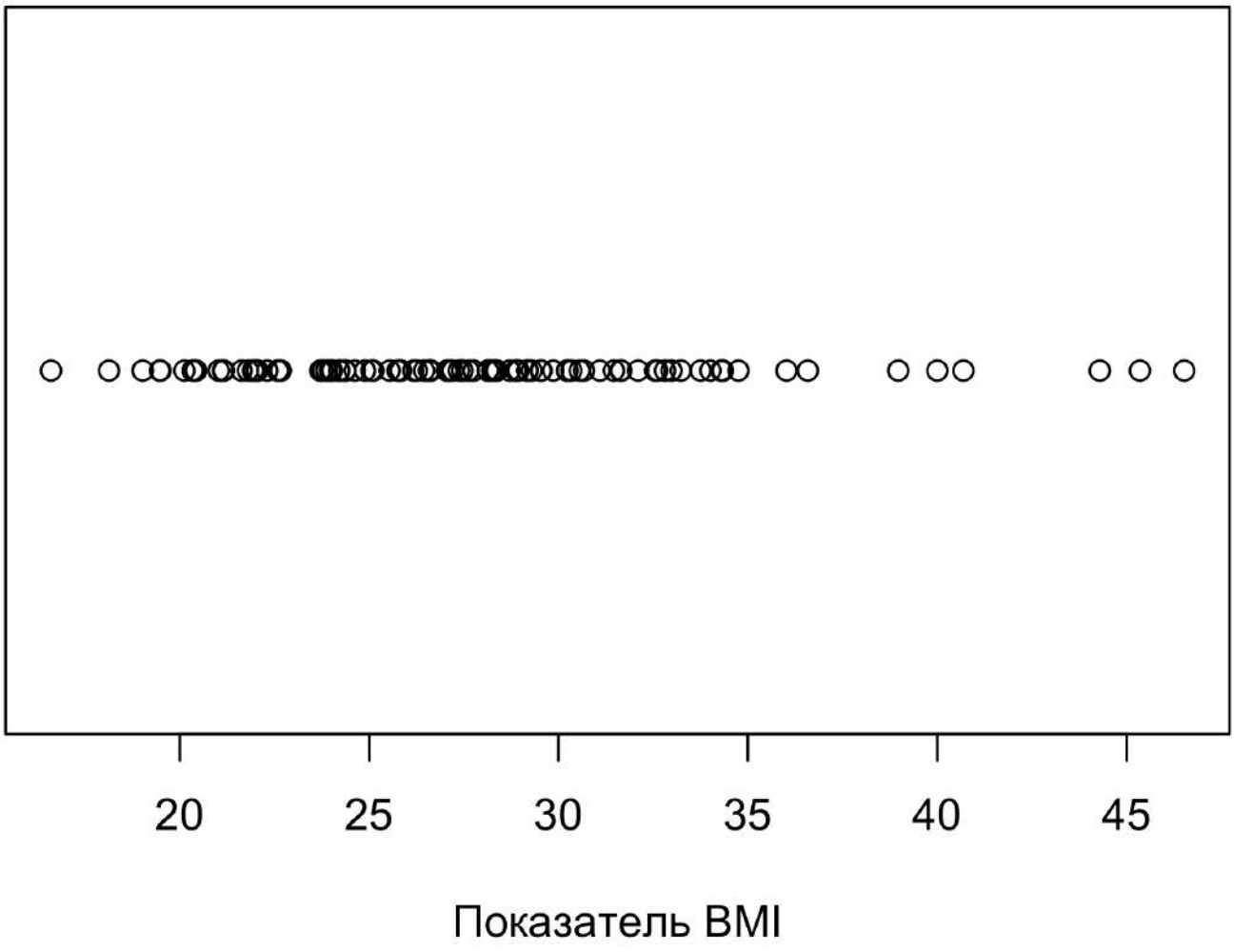
Вывод по boxplot:
Видим разную длину «ящика» для разных возрастных категорий. Можем сделать визуальный вывод о том, что возрастная категория «40-44» (самый длинный ящик) имеет наибольший разброс значений BMI, а возрастная категория «80 и старше» (самый короткий ящик) — наименьший разброс.

Stripchart

Ввиду большого объёма данных, метод stripchart плохо подходит для данного датасета. Однако мы можем осуществлять визуализацию для более узких объёмов, например, для первых 100 респондентов:

```
#stripchart
#Для корректного отображения сузим выборку.
#Посмотрим, например, на значения BMI для первых 100 респондентов
small_df <- df[-c(101:2500),]

#rounded_BMI <- round(smalldf, digits=1)
stripchart(small_df$BMI,
           xlab="Показатель BMI",
           pch=1)
```

График stripchart:



Вывод по stripchart:

Мы видим, что из первых 100 респондентов лишь 5 человек имеют ожирение третьей степени (BMI>40)

Задание 3. Проверить, являются ли наблюдения выбросами с точки зрения формальных статистических критериев Граббса и Q-теста Диксона. Визуализировать результаты.

Критерий Граббса

Нулевая гипотеза (H0): в данных нет выбросов

Альтернативная гипотеза (HA): в данных есть выброс

```
#Критерий Граббса
install.packages("outliers")
library("outliers")
grubbs.test(df$BMI, type = 11)
```

Результат критерия Граббса:

```
        Grubbs test for two opposite outliers

data:  df$BMI
G = 9.62962, U = 0.97733, p-value < 2.2e-16
alternative hypothesis: 12.48 and 75.82 are outliers
```

Вывод из результатов критерия Граббса:

Минимальное и максимальное значения BMI (12.48 и 75.82) являются выбросами.

Q-тест Диксона

Нулевая гипотеза (H0): максимум/минимум не является выбросом.

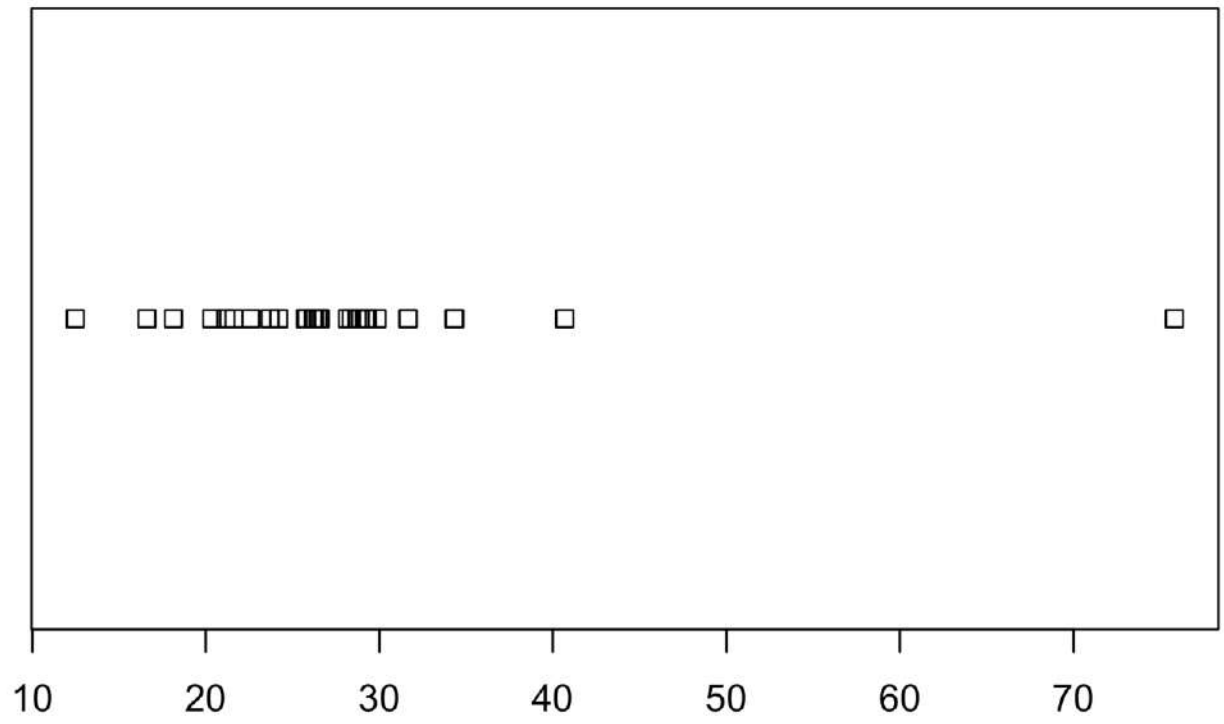
```
#Q-тест Диксона
#Применяется на небольших объёмах данных. Рассмотрим 30 наблюдений.
#Чтобы соотнести их с результатами критерия Граббса,
#рассмотрим BMI первых 28 респондентов и добавим к ним выбросы критерия Граббса
small_BMI <- c(df$BMI[1:28], 12.48, 75.82)
dixon.test(small_BMI)
dixon.test(small_BMI,opposite=TRUE)
```

Результаты Q-теста Диксона:

```
        Dixon test for outliers

data:  small_BMI
Q = 0.71902, p-value < 2.2e-16
alternative hypothesis: highest value 75.82 is an outlier

> dixon.test(small_BMI,opposite=TRUE)

        Dixon test for outliers

        Dixon test for outliers

data:  small_BMI
Q = 0.25846, p-value = 0.4936
alternative hypothesis: lowest value 12.48 is an outlier
```

Вывод из результатов Q-теста Диксона:
Значения 12.48 и 75.82 также как и по критерию Граббса являются выбросами.

Визуализация результатов с помощью stripchart(small_BMI) (видны выбросы 12.48 и 75.82, которые показали критерии):



Задание 4. Воспользоваться инструментами для заполнения пропусков в данных. Пропуски внести вручную и сравнить результаты заполнения с истинными значениями.

```
#Воспользоваться инструментами для заполнения пропусков в данных
#Создадим копию датафрейма и вручную внесём 10 пропусков в столбец BMI:
df_testing <- df
right_vals <- df_testing$BMI[c(1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101)]
df_testing$BMI[c(1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101)] = NaN
```

Воспользуемся тремя методами: заполнение пропусков средним значением в столбце, медианным значением в столбце и методом mice (многомерная оценка цепными уравнениями). Чтобы сравнить эффективность заполнения, сравним средний модуль разности между элементами вектора из «правильных» значений и вектора из значений, полученного каждым из методов.

Через среднее значение:

```
df_testing_mean <- df_testing
df_testing_mean$BMI[is.na(df_testing_mean$BMI)] <-
  mean(df_testing_mean$BMI, na.rm = T)
mean_vals <- df_testing_mean$BMI[c(1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101)]
mean(abs(right_vals-mean_vals))
```

```
> mean(abs(right_vals-mean_vals))
[1] 6.698118
```

Через медианное значение:

```
df_testing_median <- df_testing
df_testing_median$BMI[is.na(df_testing_median$BMI)] <-
  median(df_testing_median$BMI, na.rm = T)
median_vals <- df_testing_median$BMI[c(1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101)]
mean(abs(right_vals-median_vals))
```

```
> mean(abs(right_vals-median_vals))
[1] 6.187273
```

Через метод mice:

```
install.packages("mice")
library("mice")
method_use <- mice(df_testing)
df_testing_2 <- complete(method_use)
mice_vals <- df_testing_2$BMI[c(1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101)]
mean(abs(right_vals-mice_vals))
```

```
> mean(abs(right_vals-mice_vals))
[1] 6.143636
```

Вывод: метод mice имеет наименьший средний модуль отклонения от правильных значений. В данном случае, он наиболее эффективен.

Задание 5. Сгенерировать данные из нормального распределения с различными параметами и провести анализ с помощью графиков эмпирических функций распределений, квантилей, метода огибающих, а также стандартных процедур проверки гипотез о нормальности (критерии Колмогорова-Смирнова, Шапиро-Уилка, Андерсона-Дарлинга, Крамера фон Мизеса, Колмогорова-Смирнова в модификации Лиллиефорса и Шапиро-Франсия). Рассмотреть выборки малого (не более 50-100 элементов и умеренного (1000-5000 наблюдений) объёмов.

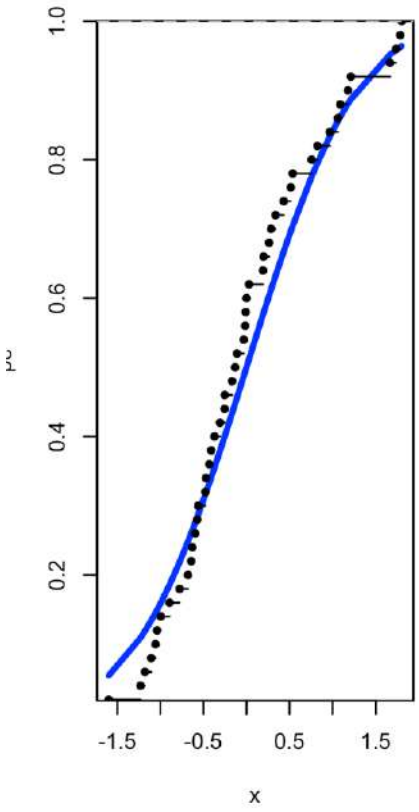
Сгенерируем 3 выборки малого и 3 выборки умеренного объёмов:

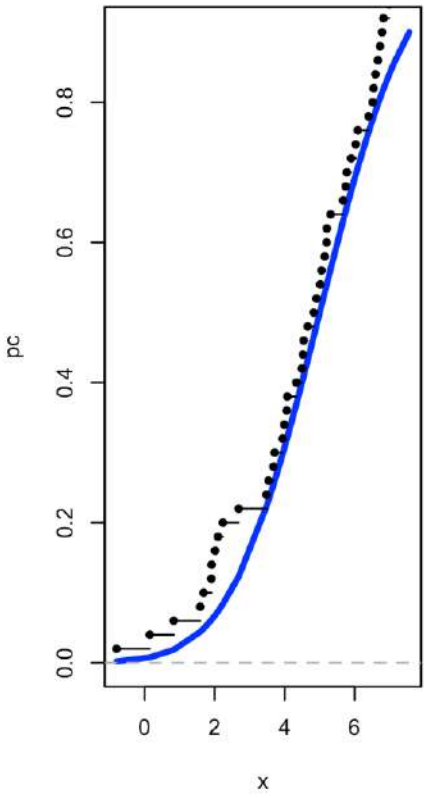
```
#Работа с произвольными данными из нормального распределения
#Малые выборки по 50 наблюдений с различными параметрами
small_1 <- rnorm(50, 0, 1)
small_2 <- rnorm(50, 5, 2)
small_3 <- rnorm(50, -5, 5)
#Умеренные выборки по 2500 наблюдений с различными параметрами
big_1 <- rnorm(2500, 0, 1)
big_2 <- rnorm(2500, 5, 2)
big_3 <- rnorm(2500, -5, 5)
```

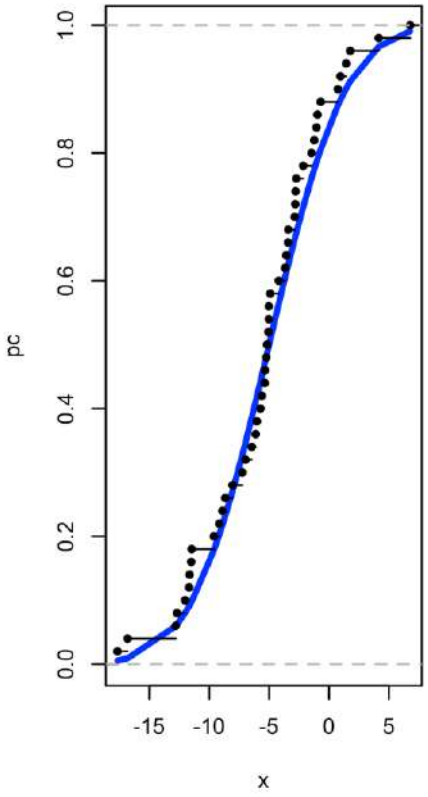
Графики эмпирических функций распределения

```
install.packages("ggplot2")
library(ggplot2)

ecdf_graph <- function(x, pc)
{
  plot(x, pc, type = "l", col = "blue", lwd = 3)
  plot(ecdf(x), add = TRUE)
}

layout(matrix(1:3, ncol = 3))
ecdf_graph(sort(small_1), pnorm(sort(small_1), mean = 0, sd = 1))
ecdf_graph(sort(small_2), pnorm(sort(small_2), mean = 5, sd = 2))
ecdf_graph(sort(small_3), pnorm(sort(small_3), mean = -5, sd = 5))

ecdf_graph(sort(big_1), pnorm(sort(big_1), mean = 0, sd = 1))
ecdf_graph(sort(big_2), pnorm(sort(big_2), mean = 5, sd = 2))
ecdf_graph(sort(big_3), pnorm(sort(big_3), mean = -5, sd = 5))
```

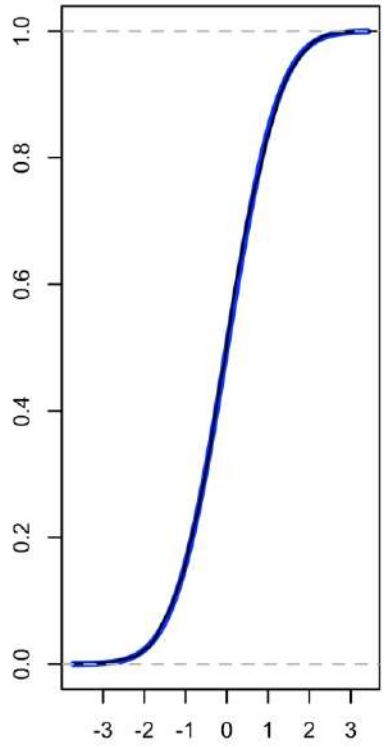
Для малых выборок:

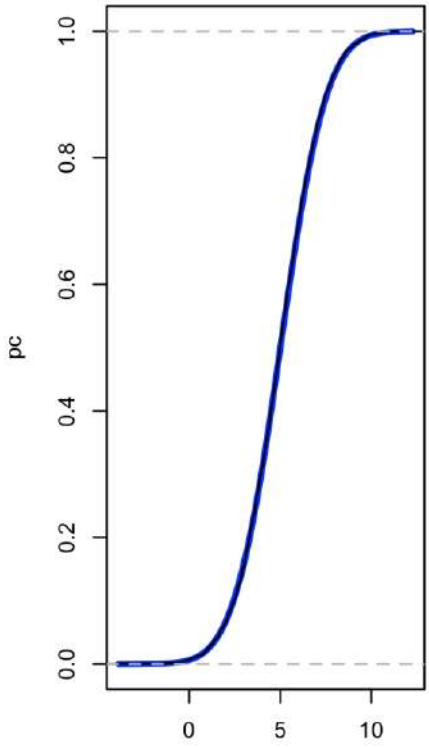


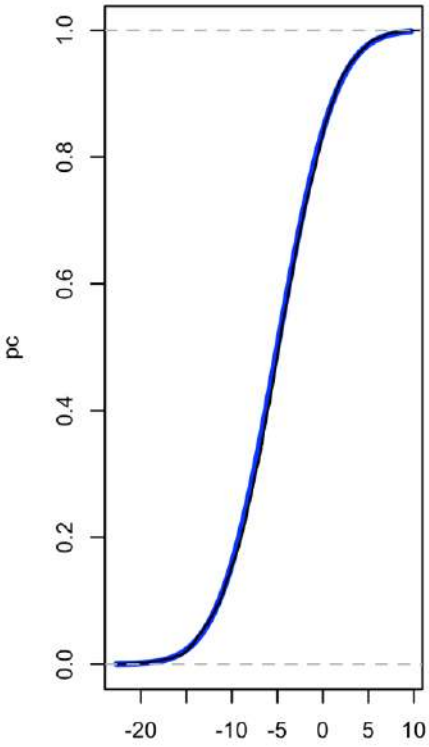




Для умеренных выборок:



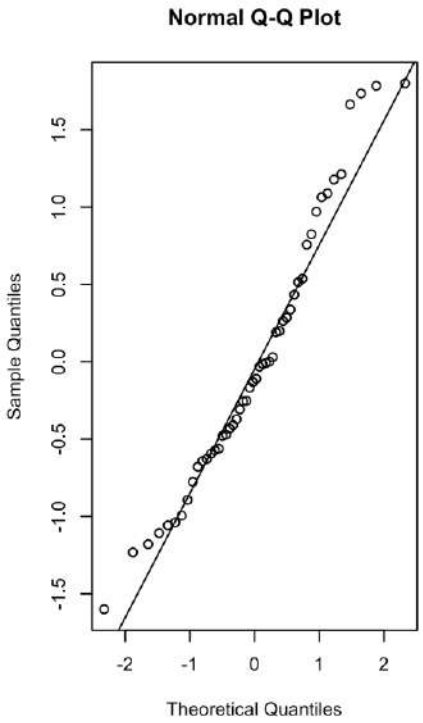


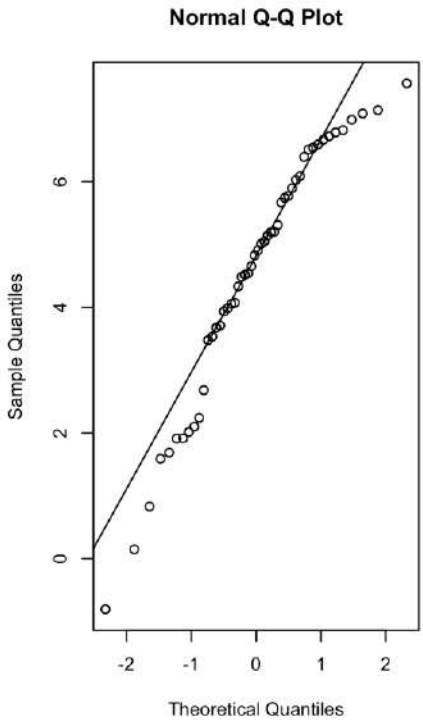


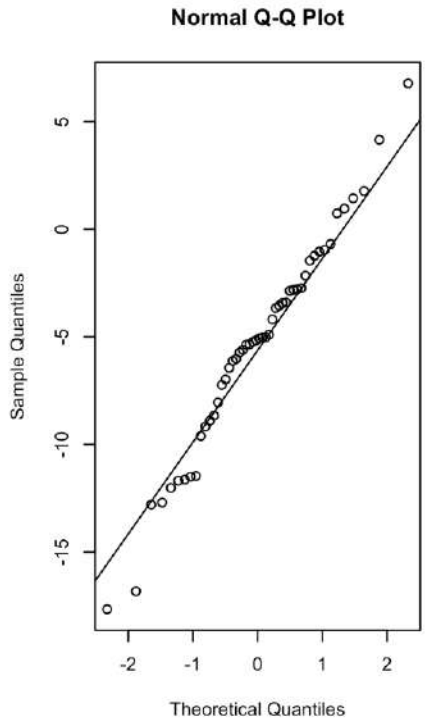
Графики квантилей

```
#Графики квантилей
quantile <- function(data){
  qqnorm(data)
  qqline(data)
}
quantile(small_1)
quantile(small_2)
quantile(small_3)
quantile(big_1)
quantile(big_2)
quantile(big_3)
```

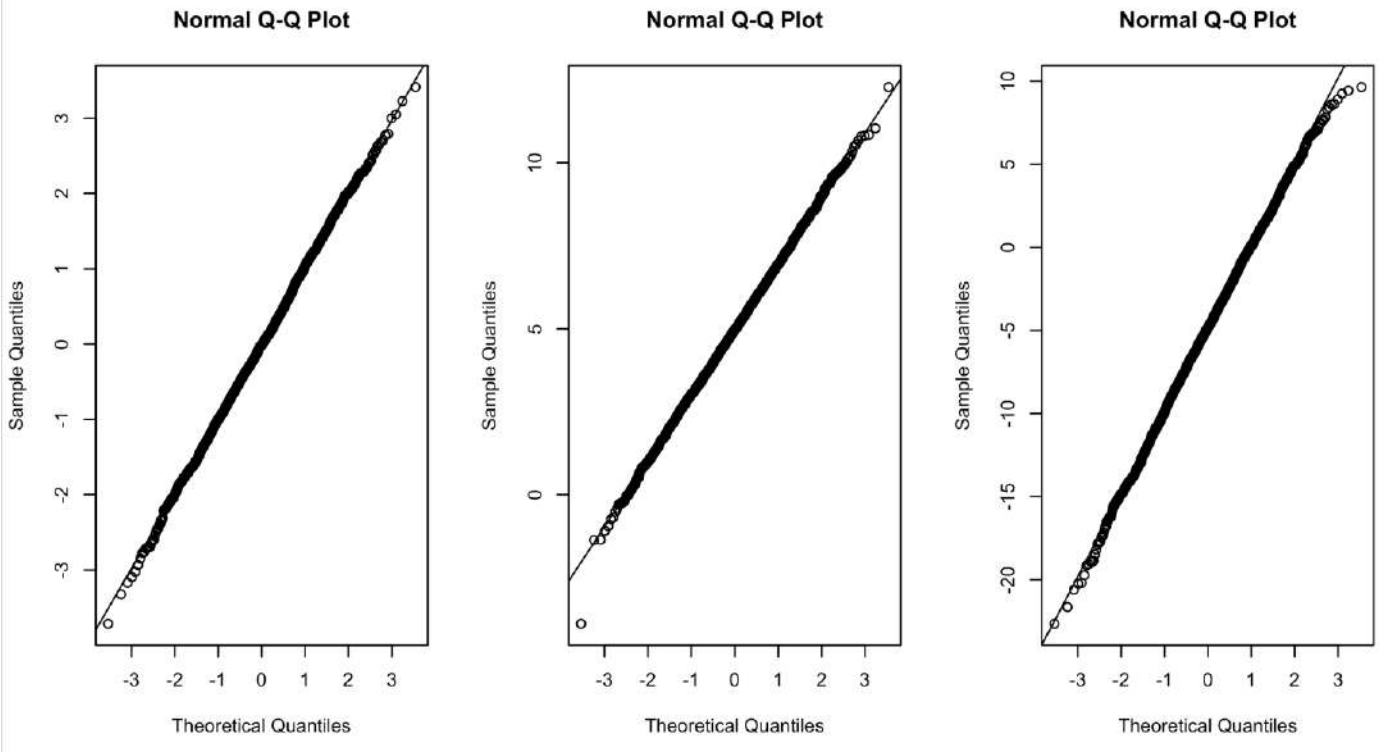
Для малых выборок:







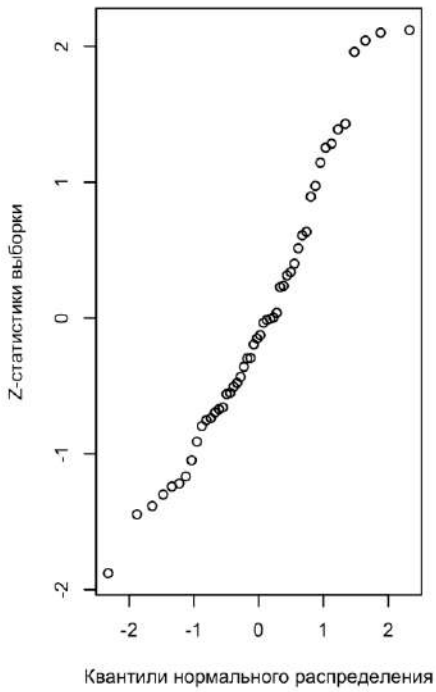
Для умеренных выборок:

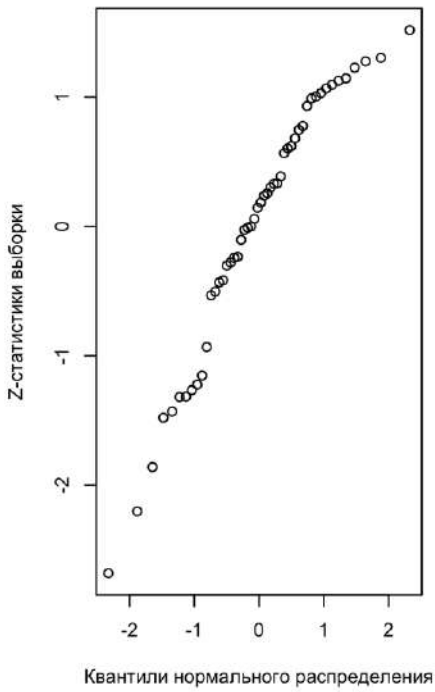


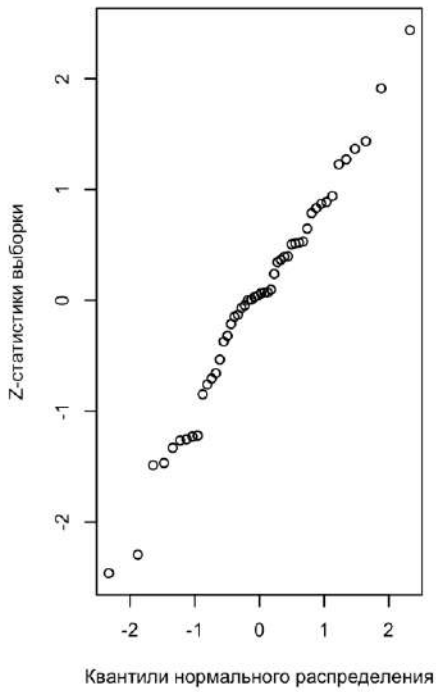
Метод огибающих

```r
#Метод огибающих
envelmet <- function(data){
  stand_data <- (data-mean(data))/sqrt(var(data))
  data.qq <- qqnorm(stand_data, plot.it = FALSE)
  data.qq <- lapply(data.qq, sort)
  plot(data.qq,
       ylab = "Z-статистики выборки",
       xlab = "Квантили нормального распределения")
}
envelmet(small_1)
envelmet(small_2)
envelmet(small_3)
envelmet(big_1)
envelmet(big_2)
envelmet(big_3)
```

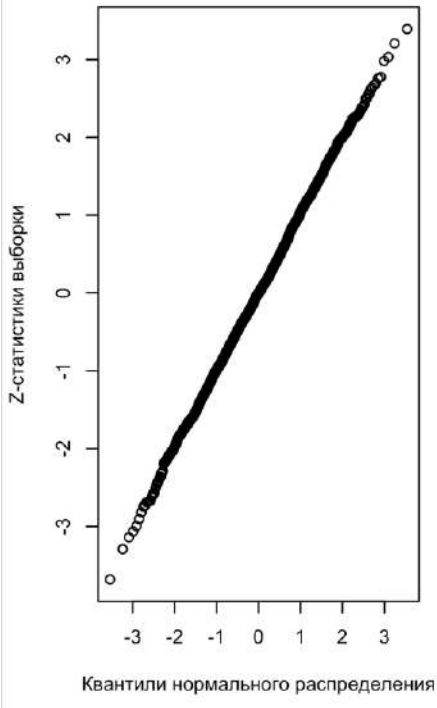
Для малых выборок:

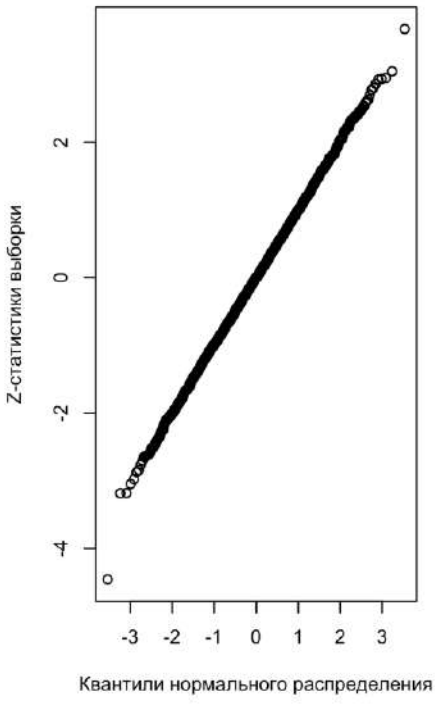


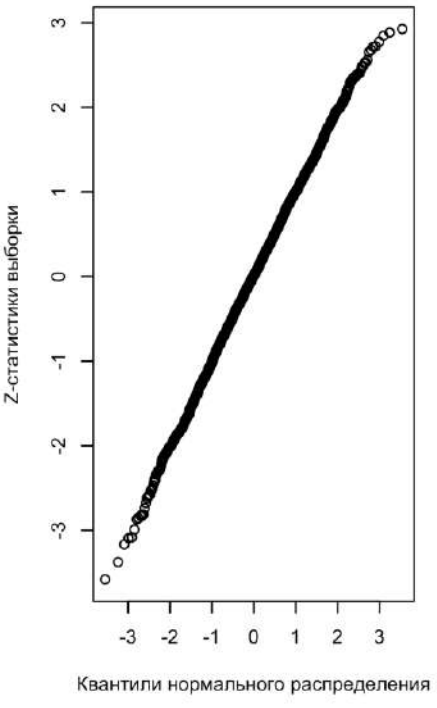




Для умеренных выборок:







Критерий Колмогорова-Смирнова

Нулевая гипотеза H0: сгенерированное распределение равно стандартному нормальному с mu=0, sigma=1.

```
#Критерий Колмогорова-Смирнова:
ks.test(small_1, "pnorm")
ks.test(small_2, "pnorm")
ks.test(small_3, "pnorm")
ks.test(big_1, "pnorm")
ks.test(big_2, "pnorm")
ks.test(big_3, "pnorm")
```

Результаты теста для малых выборок:

```
        Exact one-sample Kolmogorov-Smirnov test

data:  small_1
D = 0.10805, p-value = 0.5665
alternative hypothesis: two-sided

        Exact one-sample Kolmogorov-Smirnov test

data:  small_2
D = 0.88419, p-value = 8.882e-16
alternative hypothesis: two-sided

        Exact one-sample Kolmogorov-Smirnov test

data:  small_3
D = 0.76447, p-value = 8.882e-16
alternative hypothesis: two-sided
```

Результаты теста для умеренных выборок:

```
        Asymptotic one-sample Kolmogorov-Smirnov test

data:  big_1
D = 0.017841, p-value = 0.4038
alternative hypothesis: two-sided

        Asymptotic one-sample Kolmogorov-Smirnov test

data:  big_2
D = 0.91322, p-value < 2.2e-16
alternative hypothesis: two-sided

        Asymptotic one-sample Kolmogorov-Smirnov test

data:  big_3
D = 0.69742, p-value < 2.2e-16
alternative hypothesis: two-sided
```

Вывод: нулевая гипотеза принимается лишь для выборок small_1 и big_1 (так как p-value>0.05), которые как раз и были сгенерированы с параметрами стандартного нормального распределения. Для остальных выборок (с другими параметрами) гипотеза отклоняется (p-value<0.05).

Критерий Шапиро-Уилка

Нулевая гипотеза H0: сгенерированные данные распределены нормально

```
install.packages("nortest")
library(nortest)

#Критерий Шапиро-Уилка
shapiro.test(small_1)
shapiro.test(small_2)
shapiro.test(small_3)
shapiro.test(big_1)
shapiro.test(big_2)
shapiro.test(big_3)
```

Результаты теста для малых и умеренных выборок:

```
        Shapiro-Wilk normality test

data:  small_1
W = 0.96181, p-value = 0.1059

        Shapiro-Wilk normality test

data:  small_2
W = 0.97762, p-value = 0.4566

        Shapiro-Wilk normality test

data:  small_3
W = 0.98278, p-value = 0.6737

        Shapiro-Wilk normality test

data:  big_1
W = 0.99951, p-value = 0.8022

        Shapiro-Wilk normality test

data:  big_2
W = 0.99959, p-value = 0.9116

        Shapiro-Wilk normality test

data:  big_3
W = 0.99896, p-value = 0.1472
```

Вывод: p-value>0.05 для всех выборок, соответственно мы можем принять нулевую гипотезу для каждой из выборок.

Критерий Андерсона-Дарлинга

Нулевая гипотеза H0: сгенерированные данные распределены нормально

```
#Критерий Андерсона-Дарлинга
ad.test(small_1)
ad.test(small_2)
ad.test(small_3)
ad.test(big_1)
ad.test(big_2)
ad.test(big_3)
```

Результаты теста для малых и умеренных выборок:

```
        Anderson-Darling normality test

data:  small_1
A = 0.57841, p-value = 0.1258

        Anderson-Darling normality test

data:  small_2
A = 0.30574, p-value = 0.5542

        Anderson-Darling normality test

data:  small_3
A = 0.40457, p-value = 0.3416

        Anderson-Darling normality test

data:  big_1
A = 0.4241, p-value = 0.3179

        Anderson-Darling normality test

data:  big_2
A = 0.10767, p-value = 0.9941

        Anderson-Darling normality test

data:  big_3
A = 0.48493, p-value = 0.227
```

Вывод: p-value>0.05 для всех выборок, соответственно мы можем принять нулевую гипотезу для каждой из выборок.

Критерий Крамера фон Мизеса

Нулевая гипотеза H0: сгенерированные данные распределены нормально

```
#Критерий Крамера фон Мизеса
cvm.test(small_1)
cvm.test(small_2)
cvm.test(small_3)
cvm.test(big_1)
cvm.test(big_2)
cvm.test(big_3)
```

Результаты теста для малых и умеренных выборок:

```
        Cramer-von Mises normality test

data:  small_1
W = 0.08931, p-value = 0.153

        Cramer-von Mises normality test

data:  small_2
W = 0.046732, p-value = 0.5526

        Cramer-von Mises normality test

data:  small_3
W = 0.073787, p-value = 0.2442

        Cramer-von Mises normality test

data:  big_1
W = 0.076206, p-value = 0.2321

        Cramer-von Mises normality test

data:  big_2
W = 0.014765, p-value = 0.9947

        Cramer-von Mises normality test

data:  big_3
W = 0.069708, p-value = 0.2821
```

Вывод: p-value>0.05 для всех выборок, соответственно мы можем принять нулевую гипотезу для каждой из выборок.

Критерий Колмогорова-Смирнова в модификации Лиллиефорса

Нулевая гипотеза H0: сгенерированные данные распределены нормально

```
#Критерий Колмогорова-Смирнова в модификации Лиллиефорса
lillie.test(small_1)
lillie.test(small_2)
lillie.test(small_3)
lillie.test(big_1)
lillie.test(big_2)
lillie.test(big_3)
```

Результаты теста для малых и умеренных выборок:

```
        Lilliefors (Kolmogorov-Smirnov) normality test

data:  small_1
D = 0.10467, p-value = 0.1864

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  small_2
D = 0.083152, p-value = 0.5249

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  small_3
D = 0.10124, p-value = 0.2259

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  big_1
D = 0.016002, p-value = 0.1254

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  big_2
D = 0.007778, p-value = 0.9717

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  big_3
D = 0.013584, p-value = 0.3194
```

Вывод: p-value>0.05 для всех выборок, соответственно мы можем принять нулевую гипотезу для каждой из выборок.

Критерий Колмогорова-Смирнова в модификации Шапиро-Франсия

Нулевая гипотеза H0: сгенерированные данные распределены нормально

```
#Критерий Колмогорова-Смирнова в модификации Шапиро-Франсия
sf.test(small_1)
sf.test(small_2)
sf.test(small_3)
sf.test(big_1)
sf.test(big_2)
sf.test(big_3)
```

Результаты теста для малых и умеренных выборок:

```
        Shapiro-Francia normality test

data:  small_1
W = 0.96789, p-value = 0.1653

        Shapiro-Francia normality test

data:  small_2
W = 0.9765, p-value = 0.3511

        Shapiro-Francia normality test

data:  small_3
W = 0.98044, p-value = 0.4856

        Shapiro-Francia normality test

data:  big_1
W = 0.99952, p-value = 0.7743

        Shapiro-Francia normality test

data:  big_2
W = 0.99943, p-value = 0.631

        Shapiro-Francia normality test

data:  big_3
W = 0.99909, p-value = 0.2053
```

Вывод: p-value>0.05 для всех выборок, соответственно мы можем принять нулевую гипотезу для каждой из выборок.

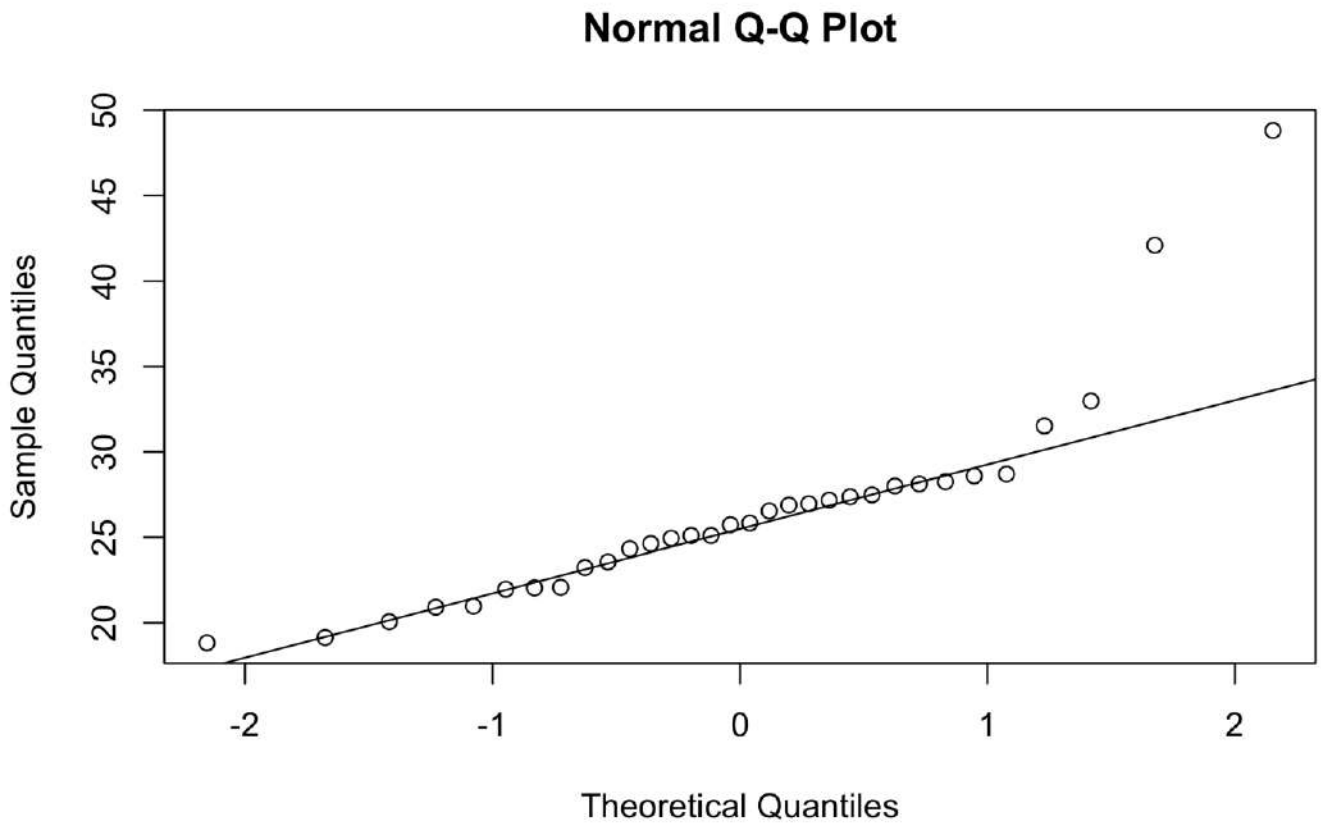
Задание 6. Продемонстрировать пример анализа данных с помощью графиков квантилей, метода огибающих, а также стандартных процедур проверки гипотез о нормальности. Рассмотреть выборки малого и умеренного объёмов.

```
#Выборка малого объёма – курящие и пьющие мужчины (32 наблюдения)
data_BMI_small <- df$BMI[df$Sex == "Male" & df$Smoking == "Yes"
                         & df$AlcoholDrinking == "Yes" ]

#Выборка умеренного объёма – столбец с данными BMI (2500 наблюдений)
data_BMI <- df$BMI
```

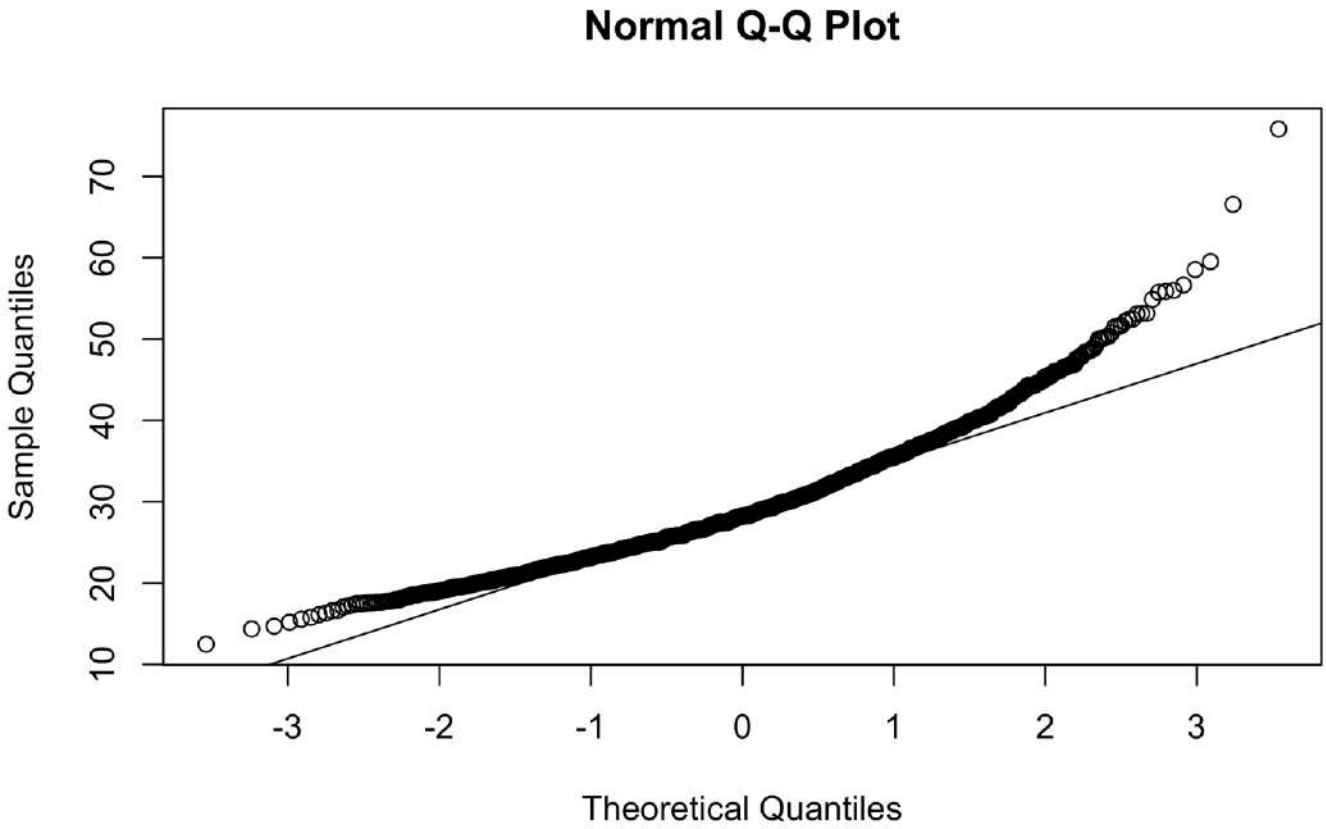
Графики квантилей

```
layout(matrix(1:2, ncol = 2))
quantile(data_BMI_small)
quantile(data_BMI)
```

Малая выборка:



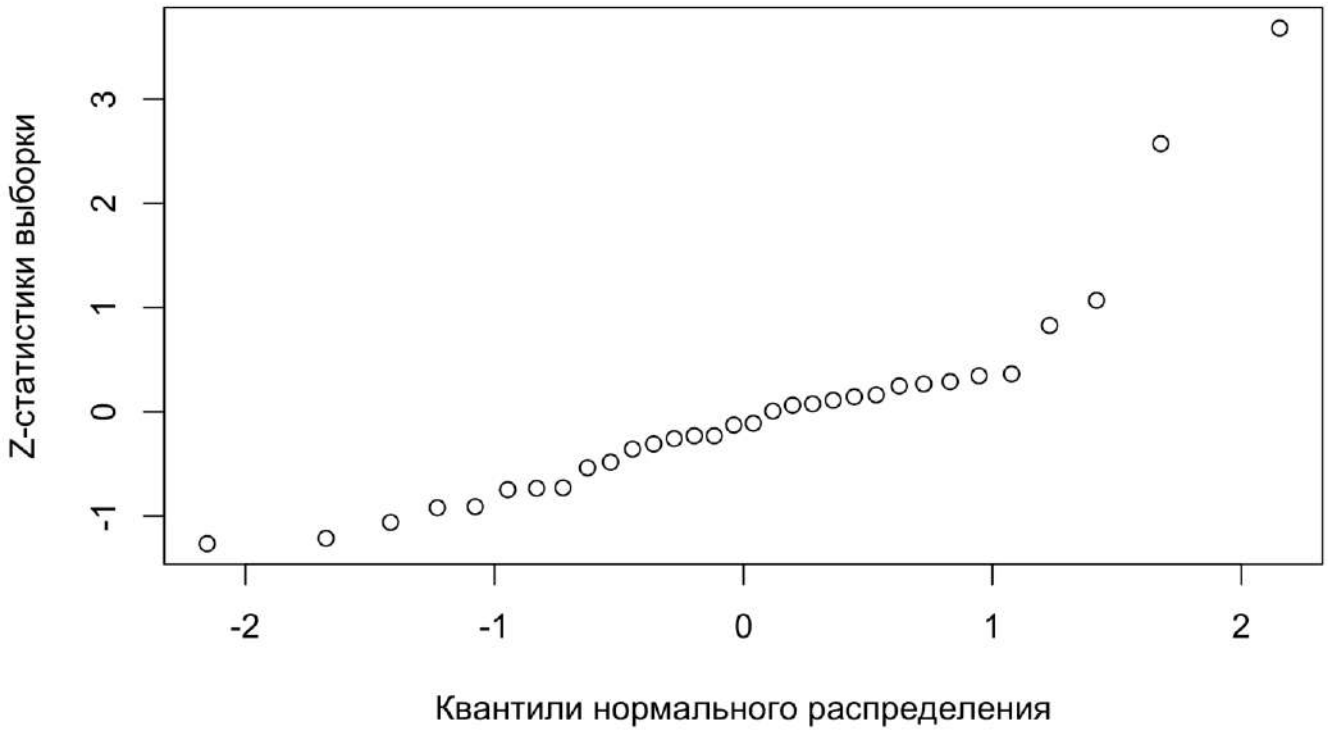
Умеренная выборка:



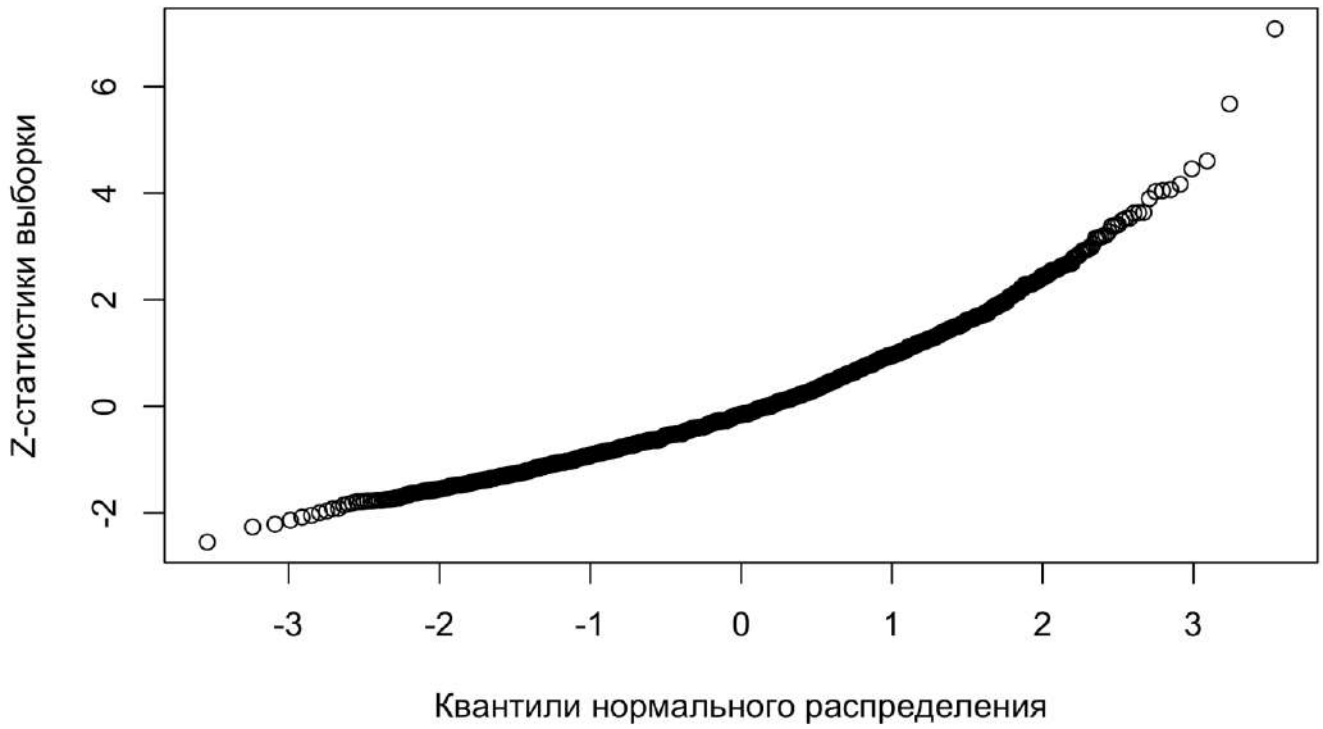
Метод огибающих

```
#Метод огибающих
envelmet(data_BMI_small)
envelmet(data_BMI)
```

Малая выборка:



Умеренная выборка:



Стандартные процедуры проверки гипотез о нормальности

```
#Применим ранее введённые критерии для проверки гипотез о нормальности
ks.test(unique(data_BMI_small), "pnorm")
ks.test(unique(data_BMI), "pnorm")

shapiro.test(data_BMI_small)
shapiro.test(data_BMI)

ad.test(data_BMI_small)
ad.test(data_BMI)

cvm.test(data_BMI_small)
cvm.test(data_BMI)

lillie.test(data_BMI_small)
lillie.test(data_BMI)

sf.test(data_BMI_small)
sf.test(data_BMI)
```

Результаты тестов для малой выборки:

```
        Exact one-sample Kolmogorov-Smirnov test

data:  unique(data_BMI_small)
D = 1, p-value = 3.331e-16
alternative hypothesis: two-sided

        Shapiro-Wilk normality test

data:  data_BMI_small
W = 0.80508, p-value = 5.086e-05

        Anderson-Darling normality test

data:  data_BMI_small
A = 1.7047, p-value = 0.0001785

        Cramer-von Mises normality test

data:  data_BMI_small
W = 0.26907, p-value = 0.0006836

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  data_BMI_small
D = 0.23322, p-value = 0.0001182

        Shapiro-Francia normality test

data:  data_BMI_small
W = 0.79083, p-value = 8.383e-05
```

Для умеренной выборки:

```
        Asymptotic one-sample Kolmogorov-Smirnov test

data:  unique(data_BMI)
D = 1, p-value < 2.2e-16
alternative hypothesis: two-sided

        Shapiro-Wilk normality test

data:  data_BMI
W = 0.95036, p-value < 2.2e-16

        Anderson-Darling normality test

data:  data_BMI
A = 25.131, p-value < 2.2e-16

        Cramer-von Mises normality test

data:  data_BMI
W = 4.2836, p-value = 7.37e-10

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  data_BMI
D = 0.076755, p-value < 2.2e-16

        Shapiro-Francia normality test

data:  data_BMI
W = 0.94986, p-value < 2.2e-16
```

Выводы:

Как для малой, так и для умеренной выборок значения каждого из тестов показали p-value<0.05, что означает, что мы можем отвергнуть гипотезу о нормальности обеих выборок.

Задание 7. Продемонстрировать применение для проверки различных гипотез и различных доверительных уровней (0.9, 0.95, 0.99) следующих критериев: а) Стьюдента (реализовать оценку мощности критериев при заданном объёме выборки); б) Уилкоксона-Манна-Уитни (ранговые); в) Фишера, Левене, Бартлетта, Флигнера-Килина (проверка гипотез об однородности дисперсий).

Критерий Стьюдента

```
#Критерий Стьюдента
#Для применения критерия необходимо,
#чтобы исходные данные имели нормальное распределение.
#Сгенерируем две нормально распределённые выборки:
norm_1 <- rnorm(50, 5, 4)
norm_2 <- rnorm(50, 2, 4)
```

Одновыборочный тест
Нулевая гипотеза H0: заданная средняя больше средней сгенерированной выборки.

```
#Одновыборочный тест
t.test(norm_1, mu = 6, conf.level = 0.9, alternative = 'greater')
t.test(norm_1, mu = 6, conf.level = 0.95, alternative = 'greater')
t.test(norm_1, mu = 6, conf.level = 0.99, alternative = 'greater')
```

Результаты теста:

```
data:  norm_1
t = -2.7722, df = 49, p-value = 0.9961
alternative hypothesis: true mean is greater than 6
90 percent confidence interval:
 3.435107      Inf
sample estimates:
mean of x
 4.253509

        One Sample t-test

data:  norm_1
t = -2.7722, df = 49, p-value = 0.9961
alternative hypothesis: true mean is greater than 6
95 percent confidence interval:
 3.197296      Inf
sample estimates:
mean of x
 4.253509

        One Sample t-test

data:  norm_1
t = -2.7722, df = 49, p-value = 0.9961
alternative hypothesis: true mean is greater than 6
99 percent confidence interval:
 2.738447      Inf
sample estimates:
mean of x
 4.253509
```

Вывод: для каждого из доверительных уровней нулевая гипотеза принимается (pvalue>0.05)

Тест №1 для двух независимых выборок
Нулевая гипотеза H0: средние двух выборок равны

```
#Гипотеза "Средняя первой выборки равна средней второй выборки"
t.test(norm_1, norm_2, conf.level = 0.9, alternative = 'two.sided')
t.test(norm_1, norm_2, conf.level = 0.95, alternative = 'two.sided')
t.test(norm_1, norm_2, conf.level = 0.99, alternative = 'two.sided')
```

Результаты теста:

```
        Welch Two Sample t-test

data:  norm_1 and norm_2
t = 3.7333, df = 95.155, p-value = 0.0003217
alternative hypothesis: true difference in means is not equal to 0
90 percent confidence interval:
 1.704777 4.437663
sample estimates:
mean of x mean of y
 4.253509  1.182289

        Welch Two Sample t-test

data:  norm_1 and norm_2
t = 3.7333, df = 95.155, p-value = 0.0003217
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.438087 4.704353
sample estimates:
mean of x mean of y
 4.253509  1.182289

        Welch Two Sample t-test

data:  norm_1 and norm_2
t = 3.7333, df = 95.155, p-value = 0.0003217
alternative hypothesis: true difference in means is not equal to 0
99 percent confidence interval:
 0.9088938 5.2335464
sample estimates:
mean of x mean of y
 4.253509  1.182289
```

Вывод: во всех случаях нулевая гипотеза отвергается (pvalue<0.05)

Тест №2 для двух независимых выборок
Нулевая гипотеза H0: средняя второй выборки больше средней первой выборки

```
#Гипотеза "Средняя второй выборки больше средней первой выборки"
t.test(norm_1, norm_2, conf.level = 0.9, alternative = 'greater')
t.test(norm_1, norm_2, conf.level = 0.95, alternative = 'greater')
t.test(norm_1, norm_2, conf.level = 0.99, alternative = 'greater')
```

Результаты теста:

```
        Welch Two Sample t-test

data:  norm_1 and norm_2
t = 3.7333, df = 95.155, p-value = 0.0001609
alternative hypothesis: true difference in means is greater than 0
90 percent confidence interval:
 2.00958     Inf
sample estimates:
mean of x mean of y
 4.253509  1.182289
        Welch Two Sample t-test

data:  norm_1 and norm_2
t = 3.7333, df = 95.155, p-value = 0.0001609
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 1.704777      Inf
sample estimates:
mean of x mean of y
 4.253509  1.182289
        Welch Two Sample t-test

data:  norm_1 and norm_2
t = 3.7333, df = 95.155, p-value = 0.0001609
alternative hypothesis: true difference in means is greater than 0
99 percent confidence interval:
 1.124684      Inf
sample estimates:
mean of x mean of y
 4.253509  1.182289
```

Вывод: во всех случаях нулевая гипотеза отвергается (pvalue<0.05)

Оценка мощности критериев при заданном объёме выборки

```
#Оценка мощности критериев при заданном объёме выборки
power.t.test(n = 50, delta = 3, sd = 4, sig.level = 0.1)
power.t.test(n = 50, delta = 3, sd = 4, sig.level = 0.05)
power.t.test(n = 50, delta = 3, sd = 4, sig.level = 0.01)
```

Результаты оценки мощности:

```
     Two-sample t test power calculation

              n = 50
          delta = 3
             sd = 4
      sig.level = 0.1
          power = 0.9811949
    alternative = two.sided

     Two-sample t test power calculation

              n = 50
          delta = 3
             sd = 4
      sig.level = 0.05
          power = 0.9602024
    alternative = two.sided

     Two-sample t test power calculation

              n = 50
          delta = 3
             sd = 4
      sig.level = 0.01
          power = 0.8665898
    alternative = two.sided
```

Результат:

Получили три значения мощности для трёх уровней значимости:

- для уровня значимости 0.1 мощность критерия — 98%
- для уровня значимости 0.05 мощность критерия — 96%
- для уровня значимости 0.01 мощность критерия — 87%

Фрагмент кода, который позволит определить объём выборки для достижения данных мощностей:

```
#Определение объёма выборки для достижения заданной мощности
power.t.test(delta = 3, sd = 4, sig.level = 0.1, power = 0.98)
power.t.test(delta = 3, sd = 4, sig.level = 0.05, power = 0.96)
power.t.test(delta = 3, sd = 4, sig.level = 0.01, power = 0.87)
```

Критерий Уилкоксона-Манна-Уитни (ранговый)

Сгенерируем две выборки из гамма-распределения:

```
#Критерий Уилкоксона-Манна-Уитни
gamma_1 <- rgamma(40, 1, 1)
gamma_2 <- rgamma(40, 2, 3)
```

Нулевая гипотеза H0: выборки распределены одинаково

```
wilcox.test(gamma_1, gamma_2, conf.level = 0.9)
wilcox.test(gamma_1, gamma_2, conf.level = 0.95)
wilcox.test(gamma_1, gamma_2, conf.level = 0.99)
```

Результаты теста:

```
        Wilcoxon rank sum exact test

data:  gamma_1 and gamma_2
W = 916, p-value = 0.268
alternative hypothesis: true location shift is not equal to 0

        Wilcoxon rank sum exact test

data:  gamma_1 and gamma_2
W = 916, p-value = 0.268
alternative hypothesis: true location shift is not equal to 0

        Wilcoxon rank sum exact test

data:  gamma_1 and gamma_2
W = 916, p-value = 0.268
alternative hypothesis: true location shift is not equal to 0
```

Вывод: в силу того, что p-value>0.05, нулевая гипотеза принимается.

Критерий Фишера

Нулевая гипотеза H0: дисперсии двух независимых выборок равны

```
#Критерий Фишера
norm_1 <- rnorm(50, 5, 4)
norm_2 <- rnorm(50, 2, 4)
norm_3 <- rnorm(50, 2, 6)

var.test(norm_1, norm_2)
var.test(norm_1, norm_3)
```

Результаты теста:

```
        F test to compare two variances

data:  norm_1 and norm_2
F = 0.91085, num df = 49, denom df = 49, p-value = 0.7451
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.5168874 1.6050948
sample estimates:
ratio of variances
         0.9108531
```

```
        F test to compare two variances

data:  norm_1 and norm_3
F = 0.44471, num df = 49, denom df = 49, p-value = 0.005357
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2523602 0.7836563
sample estimates:
ratio of variances
         0.4447063
```

Выводы:
Для выборок norm_1 и norm_2, сгенерированных с равными дисперсиями, нулевая гипотеза, ожидаемо, принимается. Для выборок norm_1 и norm_3, сгенерированных с неравными дисперсиями, нулевая гипотеза отвергается.

Критерии Левене, Бартлетта, Флигнера-Килина
Загрузим необходимые библиотеки и применим критерии к нашему датасету.
Нулевая гипотеза H0: дисперсии выборок BMI для мужчин и женщин равны.

```r
#Критерии Левене, Бартлетта, Флигнера-Килина о равенстве дисперсий
install.packages("car")
library(car)
install.packages("stats")
library(stats)

leveneTest(df$BMI, as.factor(df$Sex))
bartlett.test(df$BMI, as.factor(df$Sex))
fligner.test(df$BMI, as.factor(df$Sex))
```

Результаты теста:

```
Levene's Test for Homogeneity of Variance (center = median)
        Df F value    Pr(>F)
group    1  17.318 3.269e-05 ***
      2498

        Bartlett test of homogeneity of variances

data:  df$BMI and as.factor(df$Sex)
Bartlett's K-squared = 18.35, df = 1, p-value = 1.838e-05

        Fligner-Killeen test of homogeneity of variances

data:  df$BMI and as.factor(df$Sex)
Fligner-Killeen:med chi-squared = 20.804, df = 1, p-value = 5.087e-06
```

Вывод: согласно каждому из критериев, дисперсии данных выборок неравны (p-value<0.05)

Задание 8. Исследовать корреляционные взаимосвязи в данных с помощью коэффициентов корреляции Пирсона, Спирмена и Кендалла.

Исследуем, существует ли корреляция между физическим и ментальным здоровьем респондентов:

```
#Корреляционные взаимосвязи в данных

#Коэффициент корреляции Пирсона
cor.test(df$PhysicalHealth, df$MentalHealth)

#Коэффициент корреляции Спирмена
cor.test(df$PhysicalHealth, df$MentalHealth, method = "spearman")

#Коэффициент корреляции Кендалла
cor.test(df$PhysicalHealth, df$MentalHealth, method = "kendall")
```

Результаты тестов:

```
        Pearson's product-moment correlation

data:  df$PhysicalHealth and df$MentalHealth
t = 14.632, df = 2498, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2444540 0.3166788
sample estimates:
      cor
0.2809642

        Spearman's rank correlation rho

data:  df$PhysicalHealth and df$MentalHealth
S = 1886416209, p-value < 2.2e-16
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.2756161

        Kendall's rank correlation tau

data:  df$PhysicalHealth and df$MentalHealth
z = 14.075, p-value < 2.2e-16
alternative hypothesis: true tau is not equal to 0
sample estimates:
      tau
0.2416375
```

Вывод: в каждом из тестов p-value<0.05, следовательно, корреляции между физическим и ментальным здоровьем респондентов не обнаружено.

Задание 9. Продемонстрировать использование методов хи-квадрат, точного теста Фишера, теста МакНемара, Кохрана-Мантеля-Хензеля.

Составим матрицу — количество курящих и пьющих мужчин и женщин:

```
#№9
men_smoking <- nrow(df[df$Sex == "Male" & df$Smoking == "Yes",])
men_drinking <- nrow(df[df$Sex == "Male" & df$AlcoholDrinking == "Yes",])
women_smoking <- nrow(df[df$Sex == "Female" & df$Smoking == "Yes",])
women_drinking <- nrow(df[df$Sex == "Female" & df$AlcoholDrinking == "Yes",])

bad_habit_matrix <-
  matrix(c(men_smoking, men_drinking, women_smoking, women_drinking),
         nrow = 2, byrow = TRUE,
         dimnames = list(c("Мужчины", "Женщины"), c("Курящие", "Пьющие")))
```

```
> bad_habit_matrix
         Курящие Пьющие
Мужчины     534     47
Женщины     562     47
```

Метод хи-квадрат

Нулевая гипотеза H0: количество курящих и пьющих людей распределено одинаково среди мужчин и женщин.

```
#Метод хи-квадрат
chisq.test(bad_habit_matrix)
```

```
        Pearson's Chi-squared test with Yates' continuity correction

data:  bad_habit_matrix
X-squared = 0.01697, df = 1, p-value = 0.8964
```

Вывод: в силу того, что p-value>0.05, мы принимаем нулевую гипотезу о равенстве распределений.

Точный тест Фишера

Проверим справедливость той же гипотезы с помощью теста Фишера:

```
#Точный тест Фишера
fisher.test(bad_habit_matrix)
```

```
data:  bad_habit_matrix
p-value = 0.8305
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.6092203 1.4820723
sample estimates:
odds ratio
 0.9501971
```

Вывод: в силу того, что p-value>0.05, мы снова можем сказать, что разница между мужским и женским полом статистически не значима.

Тест МакНемара

Составим таблицу сопряжённости по параметрам “Smoking” и “AlcoholDrinking”:

```
mcnemar_data <- df
#Составим таблицу сопряжённости по параметрам "Smoking" и "AlcoholDrinking":
ct <- table(mcnemar_data[,c("Smoking","AlcoholDrinking")])
ct
```

```
> ct
       AlcoholDrinking
Smoking   No  Yes
    No  1376   28
    Yes 1030   66
```

Нулевая гипотеза H0: курение не оказывает влияния на тот факт, что человек употребляет алкоголь.

Результат теста:

```
#Тест МакНемара
mcnemar.test(ct)
```

```
        McNemar's Chi-squared test with continuity correction

data:  ct
McNemar's chi-squared = 947.07, df = 1, p-value < 2.2e-16
```

Вывод: в силу того, что p-value<0.05, нулевая гипотеза отклоняется. Таким образом, курение побуждает респондентов к употреблению алкоголя.

Тест Кохрана-Мантеля-Хензеля

Для корректного применения теста составим несколько bad_habit_matrix матриц (введённых ранее) для трёх произвольных возрастных групп:

```
#Тест Кохрана-Мантеля-Хензеля
#Составим таблицы из курящих/пьющих мужчин/женщин по трём возрастным группам
men_smoking_18 <- nrow(df[df$Sex == "Male" & df$Smoking == "Yes" & df$AgeCategory == "18-24",])
men_drinking_18 <- nrow(df[df$Sex == "Male" & df$AlcoholDrinking == "Yes" & df$AgeCategory == "18-24",])
women_smoking_18 <- nrow(df[df$Sex == "Female" & df$Smoking == "Yes" & df$AgeCategory == "18-24",])
women_drinking_18 <- nrow(df[df$Sex == "Female" & df$AlcoholDrinking == "Yes" & df$AgeCategory == "18-24",])

men_smoking_40 <- nrow(df[df$Sex == "Male" & df$Smoking == "Yes" & df$AgeCategory == "40-44",])
men_drinking_40 <- nrow(df[df$Sex == "Male" & df$AlcoholDrinking == "Yes" & df$AgeCategory == "40-44",])
women_smoking_40 <- nrow(df[df$Sex == "Female" & df$Smoking == "Yes" & df$AgeCategory == "40-44",])
women_drinking_40 <- nrow(df[df$Sex == "Female" & df$AlcoholDrinking == "Yes" & df$AgeCategory == "40-44",])

men_smoking_60 <- nrow(df[df$Sex == "Male" & df$Smoking == "Yes" & df$AgeCategory == "60-64",])
men_drinking_60 <- nrow(df[df$Sex == "Male" & df$AlcoholDrinking == "Yes" & df$AgeCategory == "60-64",])
women_smoking_60 <- nrow(df[df$Sex == "Female" & df$Smoking == "Yes" & df$AgeCategory == "60-64",])
women_drinking_60 <- nrow(df[df$Sex == "Female" & df$AlcoholDrinking == "Yes" & df$AgeCategory == "60-64",])
```

```
three_dim_table <-
  array(c(men_smoking_18, men_drinking_18, women_smoking_18, women_drinking_18,
          men_smoking_40, men_drinking_40, women_smoking_40, women_drinking_40,
          men_smoking_60, men_drinking_60, women_smoking_60, women_drinking_60),
        dim = c(2, 2, 3),
        dimnames = list(
          Sex = c("Мужчины", "Женщины"),
          Habit = c("Курящие", "Пьющие"),
          AgeCatregory = c("18-24", "40-44", "60-64")))
```

Нулевая гипотеза H0: количество курящих и пьющих людей распределено одинаково среди мужчин и женщин.

```
mantelhaen.test(three_dim_table)
```

```
        Mantel-Haenszel chi-squared test with continuity correction

data:  three_dim_table
Mantel-Haenszel X-squared = 0.81069, df = 1, p-value = 0.3679
alternative hypothesis: true common odds ratio is not equal to 1
95 percent confidence interval:
 0.6887979 3.9510242
sample estimates:
common odds ratio
         1.649684
```

Вывод: в силу того, что p-value>0.05, нулевая гипотеза принимается (как в случае с методом хи-квадрат и точным тестом фишера, где составлялась таблица 2x2)

Задание 10. Проверить наличие мультиколлинеарности в данных с помощью корреляционной матрицы и фактора инфляции дисперсии.

Корреляционная матрица

```
install.packages("corrplot")
library(corrplot)

M <- cor(df[,c(2, 5, 6, 12)])
corrplot(M, method = 'number')
```

```
                     BMI PhysicalHealth MentalHealth   SleepTime
BMI           1.00000000     0.10170746   0.04398601 -0.03768718
PhysicalHealth 0.10170746     1.00000000   0.28096416 -0.06655531
MentalHealth   0.04398601     0.28096416   1.00000000 -0.13263195
SleepTime     -0.03768718    -0.06655531  -0.13263195  1.00000000
```

Вывод: согласно корреляционной матрица, мультиколлинеарности в данных не наблюдается (корреляционные значения близки к нулю).

Фактор инфляции дисперсии

```
#Фактор инфляции дисперсии
vif(lm(BMI ~ PhysicalHealth + MentalHealth + SleepTime, data = df))
```

```
PhysicalHealth   MentalHealth      SleepTime
      1.086737       1.101296       1.018872
```

Вывод: видим отсутствие мультиколлинеарности значения индекса массы тела и параметров PhysicalHealth, MentalHealth, SleepTime (на высокую корреляцию указывают значения больше 10).

Задание 11. Исследовать зависимости в данных с помощью дисперсионного анализа.

Одним из положений дисперсионного анализа является нормальное распределение значений изучаемого признака в генеральной совокупности. Как было показано выше, численние данные из изначально загруженного датасета “Personal key indicators of heart disease” имеют распределение, отличное от нормального, поэтому не подходят для дисперсионного анализа.
Поэтому, чтобы реализовать основные методы дисперсионного анализа, рассмотрим встроенный в R набор данных ToothGrowth, содержащий данные о влиянии витамина С на рост зубов свинок:

```
#№11 Дисперсионный анализ
tooth_data <- ToothGrowth
str(tooth_data)

shapiro.test(tooth_data$len)
```

```
'data.frame':   60 obs. of  3 variables:
 $ len : num  4.2 11.5 7.3 5.8 6.4 10 11.2 11.2 5.2 7 ...
 $ supp: Factor w/ 2 levels "OJ","VC": 2 2 2 2 2 2 2 2 2 2 ...
 $ dose: num  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
> shapiro.test(tooth_data$len)

        Shapiro-Wilk normality test

data:  tooth_data$len
W = 0.96743, p-value = 0.1091
```

В столбце с данными содержится дозировка витамина С (dose: 0.5, 1, 2 мг/день) и тип введения витамина С в организм свинки (supp: апельсиновый сок (OJ) или аскорбиновая кислота (VC)).
Согласно критерию Шапиро-Уилка, данные из столбца len распределены нормально, так что мы можем использовать датасет для дисперсионного анализа.

Однофакторный дисперсионный анализ

Нулевая гипотеза H0: дозировка витамина C не влияет на длину зубов.

```
#Однофакторный дисперсионный анализ
one_factor <- aov(len ~ as.factor(dose), data=ToothGrowth)
summary(one_factor)
```

Результат анализа:

```
                Df Sum Sq Mean Sq F value   Pr(>F)
as.factor(dose)  2   2426    1213   67.42 9.53e-16 ***
Residuals       57   1026      18
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Вывод:

В силу того, что p-value=9.53e-16<0.05, мы можем отклонить нулевую гипотезу и принять альтернативную — дозировка витамина C влияет на длину зубов.

Однако, как было указано в описании данных, у нас есть 3 вида дозировки витамина C. Действительно ли между всеми видами дозировки есть отличия? Чтобы ответить на этот вопрос, воспользуемся тестом Тьюки, который реализует их попарное сравнение:

Нулевая гипотеза H0: между группами нет различий

```
TukeyHSD(one_factor)

  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = len ~ as.factor(dose), data = ToothGrowth)

$`as.factor(dose)`
        diff       lwr       upr    p adj
1-0.5  9.130  5.901805 12.358195 0.00e+00
2-0.5 15.495 12.266805 18.723195 0.00e+00
2-1    6.365  3.136805  9.593195 4.25e-05
```

Вывод: результаты показывают, что различия есть между всеми группами (p-value<0.05, так что нулевая гипотеза отвергается для каждого из попарных сравнений).

Двухфакторный дисперсионный анализ

В демонстрационном датасете есть вторая независимая переменная — тип введения витамина С в организм свинки. Она также может влиять на зависимую переменную — длину зубов. К тому же, на зависимую переменную может влиять и взаимодействие двух независимых переменных, что мы тоже учтём в следующем анализе:

```
#Двухфакторный дисперсионный анализ
two_factor <- aov(len ~ as.factor(dose) + supp + as.factor(dose):supp,
                  data=ToothGrowth)
summary(two_factor)
```

Результат анализа:

```
                     Df Sum Sq Mean Sq F value   Pr(>F)
as.factor(dose)       2 2426.4  1213.2  92.000  < 2e-16 ***
supp                  1  205.4   205.4  15.572 0.000231 ***
as.factor(dose):supp  2  108.3    54.2   4.107 0.021860 *
Residuals            54  712.1    13.2
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Вывод:

Все значения p-value<0.05. Таким образом, нулевая гипотеза отклоняется в каждом из случаев и принимается альтернативная. То есть на длину зубов влияет как дозировка витамина С, так и способ введения витамина С, так и взаимодействие дозировки и способа введения.

Можем также реализовать попарное сравнение взаимодействий независимых признаков с помощью теста Тьюки:

```
TukeyHSD(two_factor)
```

```
$`as.factor(dose):supp`
                  diff        lwr         upr     p adj
1:OJ-0.5:OJ       9.47   4.671876  14.2681238 0.0000046
2:OJ-0.5:OJ      12.83   8.031876  17.6281238 0.0000000
0.5:VC-0.5:OJ    -5.25 -10.048124  -0.4518762 0.0242521
1:VC-0.5:OJ       3.54  -1.258124   8.3381238 0.2640208
2:VC-0.5:OJ      12.91   8.111876  17.7081238 0.0000000
2:OJ-1:OJ         3.36  -1.438124   8.1581238 0.3187361
0.5:VC-1:OJ     -14.72 -19.518124  -9.9218762 0.0000000
1:VC-1:OJ        -5.93 -10.728124  -1.1318762 0.0073930
2:VC-1:OJ         3.44  -1.358124   8.2381238 0.2936430
0.5:VC-2:OJ     -18.08 -22.878124 -13.2818762 0.0000000
1:VC-2:OJ        -9.29 -14.088124  -4.4918762 0.0000069
2:VC-2:OJ         0.08  -4.718124   4.8781238 1.0000000
1:VC-0.5:VC       8.79   3.991876  13.5881238 0.0000210
2:VC-0.5:VC      18.16  13.361876  22.9581238 0.0000000
2:VC-1:VC         9.37   4.571876  14.1681238 0.0000058
```

Выводы:
Различий нет между следующими группами:
1:VC—0.5:OJ (1 мг/д+аскорбиновая кислота — 0.5 мг/д+апельсиновый сок)
2:OJ—1:OJ (2 мг/д+апельсиновый сок — 1 мг/д+апельсиновый сок)
2:VC—1:OJ (2 мг/д+аскорбиновая кислота — 1 мг/д+апельсиновый сок)
2:VC—2:OJ (2 мг/д+аскорбиновая кислота — 2 мг/д+апельсиновый сок)
Между всеми остальными группами есть статистически значимые различия.

Задание 12. Подогнать регрессионные модели (в том числе, нелинейные) к данным, а также оценить качество подобной аппроксимации.

Логистическая регрессия

Построим модель на основе данных по всем 319.795 респондентам:

```
#Логистическая регрессия

log_regr <- disease_data[,-c(5,11,12,14,16,17,18)]
log_regr[log_regr == "Yes"] <- "1"
log_regr[log_regr == "No"] <- "0"
log_regr$HeartDisease <- as.numeric(log_regr$HeartDisease)
log_regr$Smoking <- as.numeric(log_regr$Smoking)
log_regr$AlcoholDrinking <- as.numeric(log_regr$AlcoholDrinking)
log_regr$DiffWalking <- as.numeric(log_regr$DiffWalking)
log_regr$PhysicalActivity <- as.numeric(log_regr$PhysicalActivity)

input <- log_regr[,-c(8, 9)]
heart.data <- glm(formula = HeartDisease ~
                    BMI + PhysicalHealth + MentalHealth + SleepTime +
                    Smoking + AlcoholDrinking + DiffWalking + PhysicalActivity,
                  data = input, family = binomial)
summary(heart.data)
```

Результат логистической регрессии:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.2517  -0.4187  -0.3217  -0.3015   2.8016

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)      -3.2357261  0.0447147 -72.364  < 2e-16 ***
BMI               0.0063926  0.0009613   6.650 2.93e-11 ***
PhysicalHealth    0.0289791  0.0007101  40.807  < 2e-16 ***
MentalHealth     -0.0116889  0.0008123 -14.390  < 2e-16 ***
SleepTime         0.0361901  0.0040148   9.014  < 2e-16 ***
Smoking           0.6257200  0.0133594  46.838  < 2e-16 ***
AlcoholDrinking  -0.5782459  0.0318550 -18.152  < 2e-16 ***
DiffWalking       1.0041970  0.0166304  60.383  < 2e-16 ***
PhysicalActivity -0.2333632  0.0149590 -15.600  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 186906  on 319794  degrees of freedom
Residual deviance: 172011  on 319786  degrees of freedom
AIC: 172029

Number of Fisher Scoring iterations: 5
```

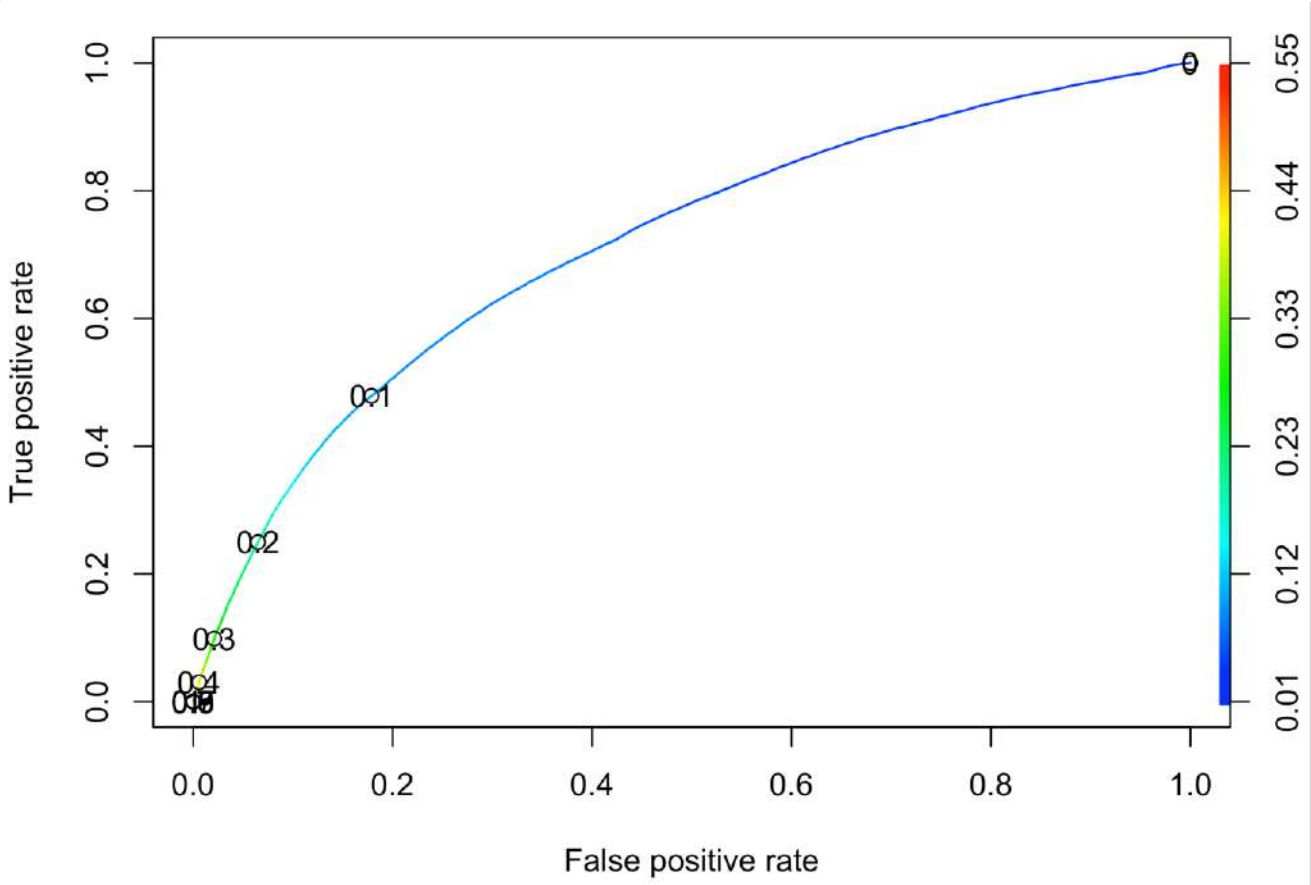
Теперь мы можем добавить в таблицу с данными столбец, где будет отображаться вероятность заболевания для каждого пациента:

```
log_regr$prob <- predict(object=heart.data, type="response")
head(log_regr$prob)
```

```
> head(log_regr$prob)
         1          2          3          4          5          6
0.05625129 0.04369963 0.10386332 0.05397134 0.22927901 0.30723077
```

Чтобы оценить качество нашей модели, построим ROC-кривую, котображает соотношение между долей объектов от общего количества носителей признака, верно классифицированных как несущие признак, и долей объектов от общего количества объектов, не несущих признака, ошибочно классифицированных как несущие признак:

```
#ROC-кривая

install.packages("ROCR")
library(ROCR)

pred_fit <- prediction(log_regr$prob, log_regr$HeartDisease)
perf_fit <- performance(pred_fit,"tpr","fpr")
plot(perf_fit, colorize=T , print.cutoffs.at = seq(0,1,by=0.1))
auc  <- performance(pred_fit, measure = "auc")
str(auc)
```



```
Formal class 'performance' [package "ROCR"] with 6 slots
  ..@ x.name      : chr "None"
  ..@ y.name      : chr "Area under the ROC curve"
  ..@ alpha.name  : chr "none"
  ..@ x.values    : list()
  ..@ y.values    :List of 1
  .. ..$ : num 0.714
  ..@ alpha.values: list()
```

Вывод:

Чем выше площадь под ROC-кривой, тем точнее наша модель может предсказывать результаты. В данном случае, показатель точности равен 0,71.