automatization\_notebook\_03

2023-10-14

# Чтение данных

В вашем варианте нужно использовать датасет framingham.

fram\_df <- read\_csv("./data/raw/framingham.csv")

## Rows: 4238 Columns: 16  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (16): male, age, education, currentSmoker, cigsPerDay, BPMeds, prevalent...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# Выведите общее описание данных

Выведем размерность и описание типов переменных. Видно, что многие переменные, должные быть фактором, на самом деле им не являются.

str(fram\_df)

## spc\_tbl\_ [4,238 × 16] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ male : num [1:4238] 1 0 1 0 0 0 0 0 1 1 ...  
## $ age : num [1:4238] 39 46 48 61 46 43 63 45 52 43 ...  
## $ education : num [1:4238] 4 2 1 3 3 2 1 2 1 1 ...  
## $ currentSmoker : num [1:4238] 0 0 1 1 1 0 0 1 0 1 ...  
## $ cigsPerDay : num [1:4238] 0 0 20 30 23 0 0 20 0 30 ...  
## $ BPMeds : num [1:4238] 0 0 0 0 0 0 0 0 0 0 ...  
## $ prevalentStroke: num [1:4238] 0 0 0 0 0 0 0 0 0 0 ...  
## $ prevalentHyp : num [1:4238] 0 0 0 1 0 1 0 0 1 1 ...  
## $ diabetes : num [1:4238] 0 0 0 0 0 0 0 0 0 0 ...  
## $ totChol : num [1:4238] 195 250 245 225 285 228 205 313 260 225 ...  
## $ sysBP : num [1:4238] 106 121 128 150 130 ...  
## $ diaBP : num [1:4238] 70 81 80 95 84 110 71 71 89 107 ...  
## $ BMI : num [1:4238] 27 28.7 25.3 28.6 23.1 ...  
## $ heartRate : num [1:4238] 80 95 75 65 85 77 60 79 76 93 ...  
## $ glucose : num [1:4238] 77 76 70 103 85 99 85 78 79 88 ...  
## $ TenYearCHD : num [1:4238] 0 0 0 1 0 0 1 0 0 0 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. male = col\_double(),  
## .. age = col\_double(),  
## .. education = col\_double(),  
## .. currentSmoker = col\_double(),  
## .. cigsPerDay = col\_double(),  
## .. BPMeds = col\_double(),  
## .. prevalentStroke = col\_double(),  
## .. prevalentHyp = col\_double(),  
## .. diabetes = col\_double(),  
## .. totChol = col\_double(),  
## .. sysBP = col\_double(),  
## .. diaBP = col\_double(),  
## .. BMI = col\_double(),  
## .. heartRate = col\_double(),  
## .. glucose = col\_double(),  
## .. TenYearCHD = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

# Очистка данных

1. Уберите переменные, в которых пропущенных значений больше 20% или уберите субъектов со слишком большим количеством пропущенных значений. Или совместите оба варианта. Напишите обоснование, почему вы выбрали тот или иной вариант:

**Обоснование**: В моем случае все было просто: переменных с такой долей NA не было вообще, поэтому я фильровал по единицам наблюдения.

1. Переименуйте переменные в человекочитаемый вид (что делать с пробелами в названиях?);
2. В соответствии с описанием данных приведите переменные к нужному типу (numeric или factor);
3. Отсортируйте данные по возрасту по убыванию;
4. Сохраните в файл outliers.csv субъектов, которые являются выбросами (например, по правилу трёх сигм) — это необязательное задание со звёздочкой;
5. Присвойте получившийся датасет переменной “cleaned\_data”.
6. Создаем вектор имен потенциально “неприкасаемых переменных”

fram\_df %>%  
 select(which(colSums(is.na(.)) > dim(.)[1] \* 0.2)) %>%  
 names

## character(0)

Переменных, где число пропущенных значений превышало бы 20% нет, засим не создаем вектор.

Оценим, какое количество и какую долю данных мы теряем при удалении переменных с числом NA равным n.

excl\_vis <- tibble(col\_count = 1:ncol(fram\_df),  
 ou\_excl = sapply(col\_count, function(x) n\_na\_excluded(fram\_df, x)),  
 ou\_excl\_perc = ou\_excl/nrow(fram\_df))

Построим график того, сколько мы теряем в абсолютных значениях при отсечке в n NA на единицу наблюдения

ggplot(excl\_vis)+  
 geom\_col(aes(x=col\_count, y=ou\_excl), fill="green", color="black")+  
 geom\_path(aes(x=col\_count, y=ou\_excl), size=0.8)+  
 labs(x = "Threshold of NAs for excluding cases",  
 y = "Number of cases with NAs\n greater or equal threshold",  
 title = "Fig. 1 Number of excluded cases \ndepending of NA numbers in case")+  
 scale\_x\_continuous(n.breaks = nrow(excl\_vis))+  
 theme\_bw()+  
 theme(axis.text.x = element\_text(size=18),  
 axis.text.y = element\_text(size=16),  
 axis.title.x = element\_text(size=24),  
 axis.title.y = element\_text(size=21),  
 plot.title = element\_text(size=25, hjust=0.5))

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

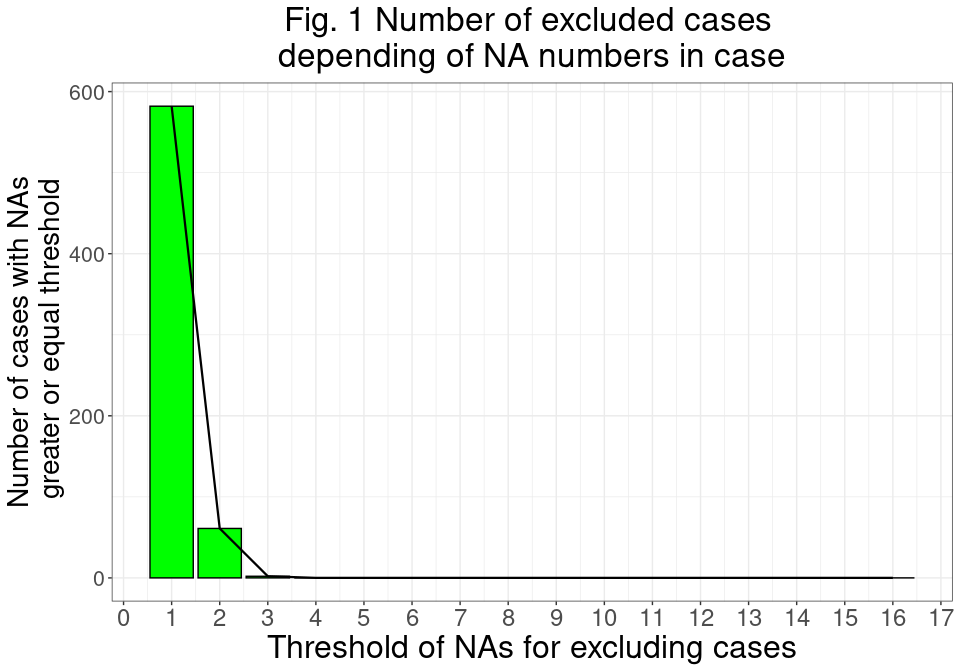
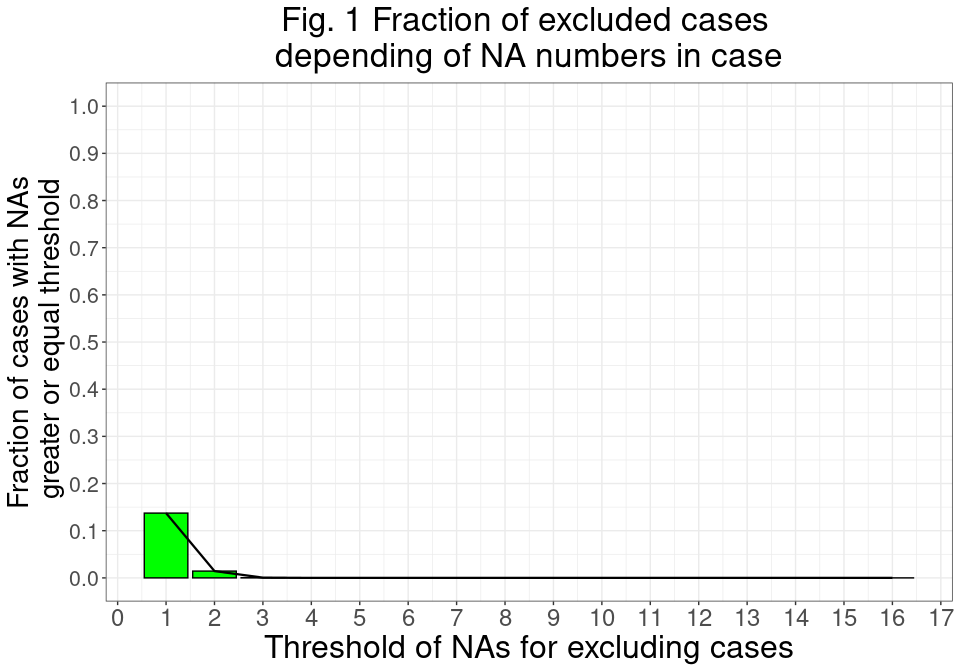


График того, какую долю респондентов мы исключаем при отсечке в n NA на единицу наблюдения

ggplot(excl\_vis)+  
 geom\_col(aes(x=col\_count, y=ou\_excl\_perc), fill="green", color="black")+  
 geom\_path(aes(x=col\_count, y=ou\_excl\_perc), size=0.8)+  
 labs(x = "Threshold of NAs for excluding cases",  
 y = "Fraction of cases with NAs\n greater or equal threshold",  
 title = "Fig. 1 Fraction of excluded cases \ndepending of NA numbers in case")+  
 scale\_x\_continuous(n.breaks = nrow(excl\_vis))+  
 scale\_y\_continuous(n.breaks = 10, limits = c(0,1))+  
 theme\_bw()+  
 theme(axis.text.x = element\_text(size=18),  
 axis.text.y = element\_text(size=16),  
 axis.title.x = element\_text(size=24),  
 axis.title.y = element\_text(size=21),  
 plot.title = element\_text(size=25, hjust=0.5))

 Видим, что при удалении случаев с одним отсутствующим значением в наблюдениях теряем около 15% оных. При удалении же случаев с двумя и более NA, теряем пренебрижимо малую часть наблюдений. Засим удалим именно их.

2-4) - На мой взгляд, все текущие переменные выглядят человекочитаемыми. Но чтобы выполнить задание по переименованию, частично расшифрую некоторые аббревиатуры и изменю стиль написания ряда переменных. Логика расшифровки аббревиатур в не вполне очевидных случаях была основана на использовании ссылки на датасет в архиве с домашним заданием. Кроме того, я позволил себе приветси переменную education к факторному типу, хотя в описании датасета на кагл она не значится. Ориентировался на данные [из этого гитхаба](https://github.com/GauravPadawe/Framingham-Heart-Study/blob/master/README.md). - На текущий момент все наши данные имеют формат numeric (double). Приведем истинно факторные переменные к факторному же типу.

# Запишем данные в промежуточный файл  
fram\_df <- fram\_df %>%  
 slice(which(rowSums(is.na(.)) < 2)) %>% # Убрать единицы наблюдения, где два или больше NA  
 rename(sex=male, # Переименовать переменные  
 cigarsPerDay = cigsPerDay,  
 bloodPressureDrugs = BPMeds,  
 strokeAnamnesis = prevalentStroke,  
 hypertAnamnesis = prevalentHyp,  
 cholTotal = totChol,  
 sysBloodPressure = sysBP,  
 diaBloodPressure = diaBP,  
 bodyMassIndex = BMI,  
 heartRate = heartRate,  
 tenYearsCHDRisk = TenYearCHD) %>%   
 mutate(sex = factor(sex, # Перевести необходимые переменные в факторы  
 labels=c("female", "male")),   
 education = factor(education,   
 labels = c("some\_high\_school", "high\_school\_GED",  
 "some\_college\_vocational\_school", "college")),  
 currentSmoker = factor(currentSmoker,   
 labels=c("no\_smoker", "smoker")),  
 bloodPressureDrugs = factor(bloodPressureDrugs,   
 labels=c("no\_medications", "medications")),  
 strokeAnamnesis = factor(strokeAnamnesis,   
 labels=c("no\_stroke", "stroke")),  
 hypertAnamnesis = factor(hypertAnamnesis,   
 labels=c("no\_hypertension", "hypertension")),  
 diabetes = factor(diabetes,   
 labels=c("no\_diabetes", "diabetes")),  
 tenYearsCHDRisk = factor(tenYearsCHDRisk, labels=c("small\_risk", "high\_risk"))) %>%  
 arrange(desc(age)) # Отсортировать по возрасту по убыванию

5-6) - Выбросы. Есть мысль, что отсеивать выбросы, не разобравшись в их структуре может быть не очень верно. Но дополнительные баллы не пахнут, засим выделяем их в отдельный датафрейм, но далее работаем с популяцией без выбросов. Использовано правило тех сигм.

outliers <- fram\_df %>%   
 filter(if\_any(.cols=is.numeric, .fns=function(x) x > mean(x, na.rm=TRUE) + 3 \* sd(x, na.rm=TRUE) | x < mean(x, na.rm=TRUE) - 3 \* sd(x, na.rm=TRUE)))

## Warning: Use of bare predicate functions was deprecated in tidyselect 1.1.0.  
## ℹ Please use wrap predicates in `where()` instead.  
## # Was:  
## data %>% select(is.numeric)  
##   
## # Now:  
## data %>% select(where(is.numeric))  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

write\_csv(outliers, "./outliers.csv")  
  
cleaned\_data <- anti\_join(fram\_df, outliers)

## Joining with `by = join\_by(sex, age, education, currentSmoker, cigarsPerDay,  
## bloodPressureDrugs, strokeAnamnesis, hypertAnamnesis, diabetes, cholTotal,  
## sysBloodPressure, diaBloodPressure, bodyMassIndex, heartRate, glucose,  
## tenYearsCHDRisk)`

# Сколько осталось переменных?

print(glue("{dim(cleaned\_data)[2]} variables retained"))

## 16 variables retained

# Сколько осталось случаев?

print(glue("{dim(cleaned\_data)[1]} cases retained"))

## 3951 cases retained

# Есть ли в данных идентичные строки?

ifelse(nrow(distinct(cleaned\_data)) == dim(cleaned\_data)[1],   
 "There are not any identical rows in dataset",   
 "There are some identical rows in dataset")

## [1] "There are not any identical rows in dataset"

# Сколько всего переменных с пропущенными значениями в данных и сколько пропущенных точек в каждой такой переменной?

vars\_with\_nas <- cleaned\_data %>%   
 select(which(colSums(is.na(.)) > 0)) %>%   
 colnames  
  
print(glue("There are {cleaned\_data %>%  
 select(vars\_with\_nas) %>%   
 summarise(across(everything(), function(x) sum(is.na(x))))} case(s) with NA in {vars\_with\_nas} "))

## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.  
## ℹ Please use `all\_of()` or `any\_of()` instead.  
## # Was:  
## data %>% select(vars\_with\_nas)  
##   
## # Now:  
## data %>% select(all\_of(vars\_with\_nas))  
##   
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

## There are 88 case(s) with NA in education   
## There are 22 case(s) with NA in cigarsPerDay   
## There are 45 case(s) with NA in bloodPressureDrugs   
## There are 9 case(s) with NA in cholTotal   
## There are 12 case(s) with NA in bodyMassIndex   
## There are 1 case(s) with NA in heartRate   
## There are 318 case(s) with NA in glucose

# Описательные статистики

## Количественные переменные

1. Рассчитайте для всех количественных переменных для каждой группы (TenYearCHD):

1.1) Количество значений;

1.2) Количество пропущенных значений;

1.3) Среднее;

1.4) Медиану;

1.5) Стандартное отклонение;

1.6) 25% квантиль и 75% квантиль;

1.7) Интерквартильный размах;

1.8) Минимум;

1.9) Максимум;

1.10) 95% ДИ для среднего - задание со звёздочкой.

statistics <- list(  
 Counts = ~ length(.x) %>% as.character(),  
 NAs = ~ sum(is.na(.x)) %>% as.character(),  
 Mean = ~ mean(.x, na.rm=TRUE) %>% round(3) %>% as.character(),  
 Median = ~ median(.x, na.rm=TRUE) %>% round(3) %>% as.character(),  
 SD = ~ sd(.x, na.rm=TRUE) %>% round(3) %>% as.character(),  
 Quantiles = ~ paste(round(quantile(.x, probs=c(0.25), na.rm=TRUE), 3),   
 round(quantile(.x, probs=c(0.75), na.rm=TRUE), 3), sep="\_"),  
 Iqr = ~ IQR(.x, na.rm=TRUE) %>% round(3) %>% as.character(),  
 Min = ~ min(.x, na.rm=TRUE) %>% round(3) %>% as.character(),  
 Max = ~ max(.x, na.rm=TRUE) %>% round(3) %>% as.character(),  
 CI95 = ~ paste(round(mean(.x, na.rm=TRUE) - 1.96 \* se(.x), 3),  
 round(mean(.x, na.rm=TRUE) + 1.96 \* se(.x), 3), sep="\_")  
)

cleaned\_data %>%   
 group\_by(tenYearsCHDRisk) %>%   
 summarise(across(is.numeric, statistics)) %>%   
 pivot\_longer(!tenYearsCHDRisk) %>%   
 separate(name, sep="\_", into=c("variable", "statistic"))

## # A tibble: 160 × 4  
## tenYearsCHDRisk variable statistic value   
## <fct> <chr> <chr> <chr>   
## 1 small\_risk age Counts 3399   
## 2 small\_risk age NAs 0   
## 3 small\_risk age Mean 48.557   
## 4 small\_risk age Median 47   
## 5 small\_risk age SD 8.344   
## 6 small\_risk age Quantiles 42\_55   
## 7 small\_risk age Iqr 13   
## 8 small\_risk age Min 32   
## 9 small\_risk age Max 70   
## 10 small\_risk age CI95 48.277\_48.838  
## # ℹ 150 more rows

## Категориальные переменные

1. Рассчитайте для всех категориальных переменных для каждой группы (TenYearCHD):

1.1) Абсолютное количество;

1.2) Относительное количество внутри группы;

1.3) 95% ДИ для доли внутри группы - задание со звёздочкой.

catStat <- function(df, factorr, grouperr){  
 df %>%   
 select({{ grouperr }}, variant = {{ factorr }}) %>%   
 mutate(variant = as.character(variant) %>% replace\_na("no\_data") %>% as.factor()) %>%   
 count({{ grouperr }}, variant) %>%   
 rename(number = n) %>%   
 group\_by({{ grouperr }}) %>%   
 mutate(proportionIntoGroup = round(number/sum(number),3),  
 proportionCI = paste(  
 round(binconf(number, sum(number), method = "wilson")[,2], 3),  
 round(binconf(number, sum(number), method = "wilson")[,3], 3),  
 sep="-"),  
 variable = factorr) %>%   
 relocate(variable, .after=1)  
}  
  
lapply(cleaned\_data %>%   
 select(where(is.factor) & !tenYearsCHDRisk) %>%   
 colnames,   
 function(x) catStat(cleaned\_data, x, tenYearsCHDRisk)) %>%   
 do.call(rbind, .) %>%   
 arrange(tenYearsCHDRisk)

## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.  
## ℹ Please use `all\_of()` or `any\_of()` instead.  
## # Was:  
## data %>% select(x)  
##   
## # Now:  
## data %>% select(all\_of(x))  
##   
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

## # A tibble: 36 × 6  
## # Groups: tenYearsCHDRisk [2]  
## tenYearsCHDRisk variable variant number proportionIntoGroup proportionCI  
## <fct> <chr> <fct> <int> <dbl> <chr>   
## 1 small\_risk sex female 1984 0.584 0.567-0.6   
## 2 small\_risk sex male 1415 0.416 0.4-0.433   
## 3 small\_risk education college 387 0.114 0.104-0.125   
## 4 small\_risk education high\_s… 1059 0.312 0.296-0.327   
## 5 small\_risk education no\_data 75 0.022 0.018-0.028   
## 6 small\_risk education some\_c… 577 0.17 0.158-0.183   
## 7 small\_risk education some\_h… 1301 0.383 0.367-0.399   
## 8 small\_risk currentSmoker no\_smo… 1729 0.509 0.492-0.525   
## 9 small\_risk currentSmoker smoker 1670 0.491 0.475-0.508   
## 10 small\_risk bloodPressur… medica… 72 0.021 0.017-0.027   
## # ℹ 26 more rows

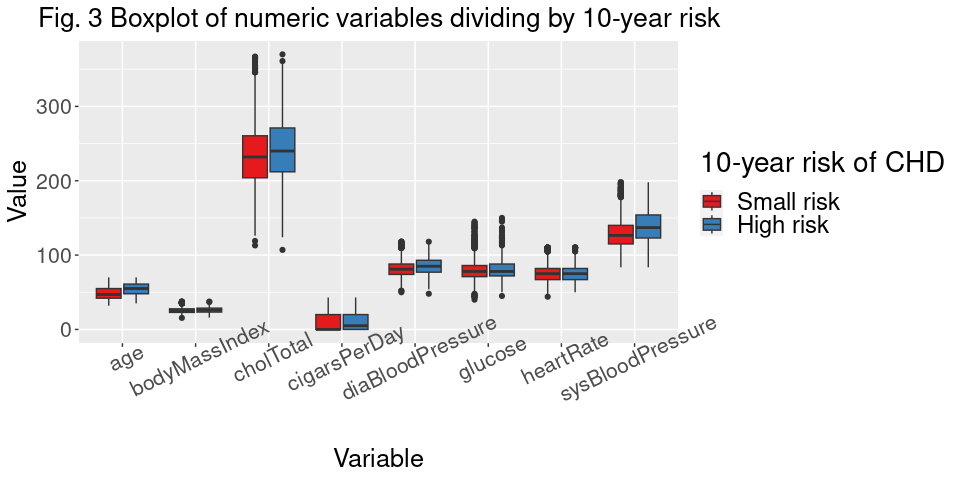
# Визуализация

## Количественные переменные

1. Для каждой количественной переменной сделайте боксплоты по группам. Расположите их либо на отдельных рисунках, либо на одном, но читаемо;
2. Наложите на боксплоты beeplots - задание со звёздочкой.
3. Раскрасьте боксплоты с помощью библиотеки RColorBrewer.

cleaned\_data\_long <- cleaned\_data %>%   
 select(where(is.numeric), tenYearsCHDRisk) %>%  
 pivot\_longer(cols=where(is.numeric), names\_to = "variable")  
  
ggplot(cleaned\_data\_long)+  
 geom\_boxplot(aes(x=variable, y=value, fill=tenYearsCHDRisk))+  
 scale\_fill\_brewer(name="10-year risk of CHD",  
 palette = "Set1",  
 labels=c("Small risk", "High risk"))+  
 labs(x = "Variable",  
 y = "Value",  
 title = "Fig. 3 Boxplot of numeric variables dividing by 10-year risk")+  
 theme(axis.text.x = element\_text(size=16, angle=26),  
 axis.text.y = element\_text(size=16),  
 axis.title.x = element\_text(size=19),  
 axis.title.y = element\_text(size=19),  
 plot.title = element\_text(size=20, hjust=0.5),  
 legend.title = element\_text(size=21),  
 legend.text = element\_text(size=18))

## Warning: Removed 362 rows containing non-finite values (`stat\_boxplot()`).



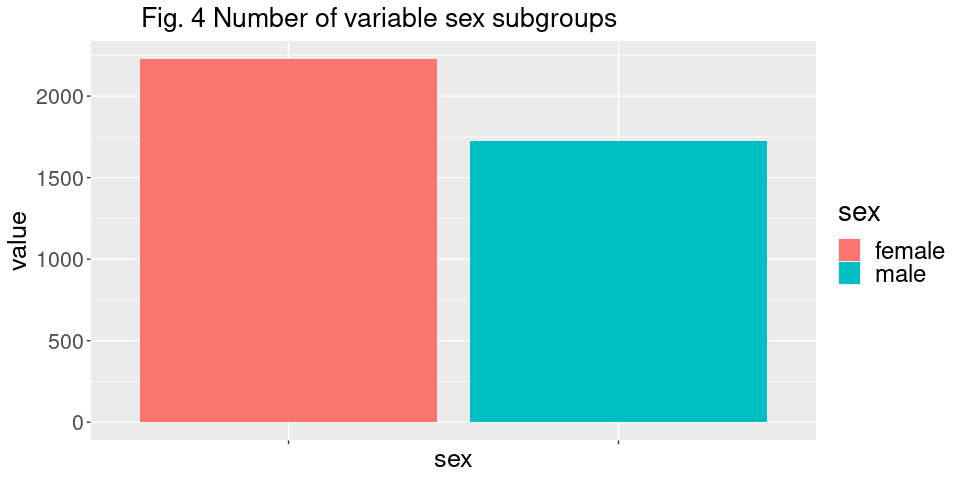
## Категориальные переменные

1. Сделайте подходящие визуализации категориальных переменных. Обоснуйте, почему выбрали именно этот тип.

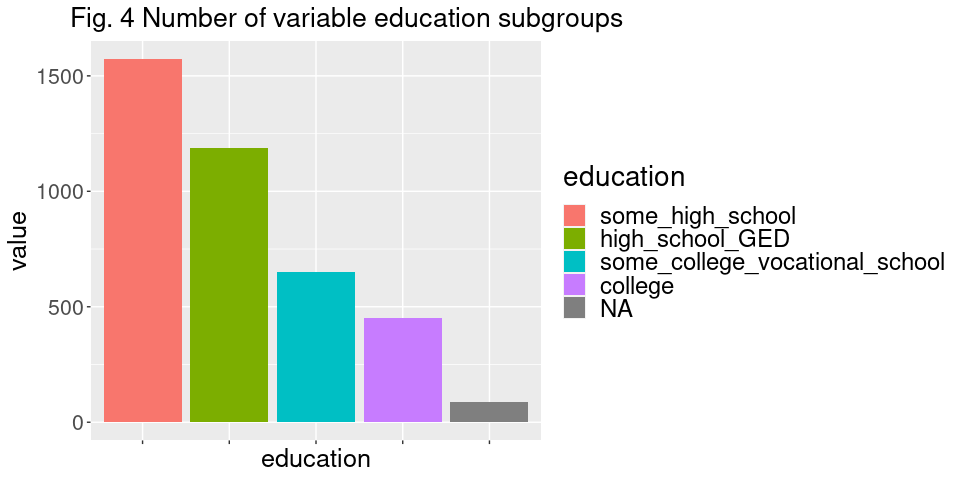
Для визуализации была выбрана столбчатая диаграмма. Она позволяет адекватно визуализировать количество единиц наблюдения для каждой из категорий фактора.

lapply(cleaned\_data %>% select(where(is.factor)) %>% colnames, function(x) barCreator(cleaned\_data, x))

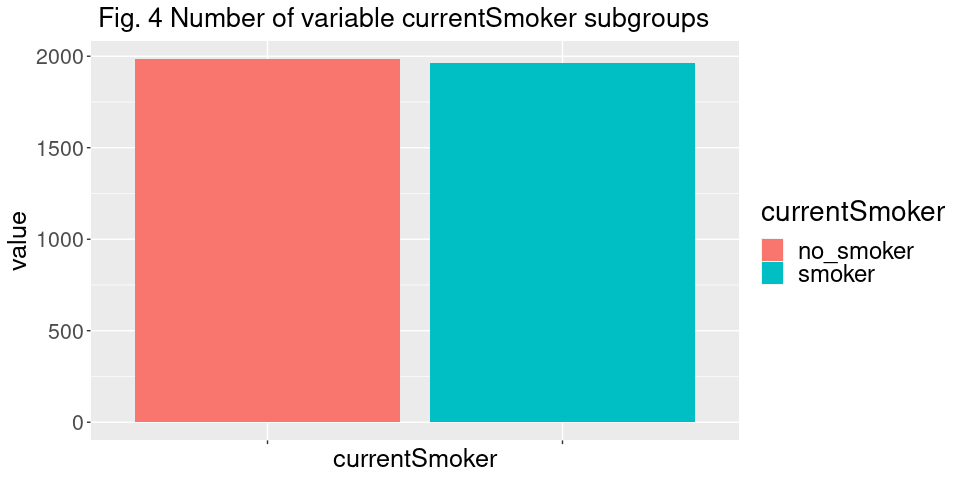
## [[1]]



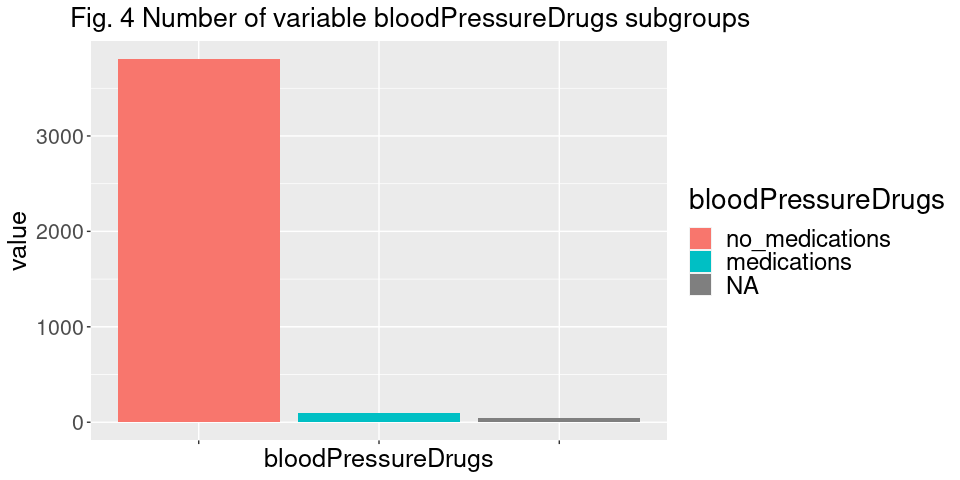
##   
## [[2]]



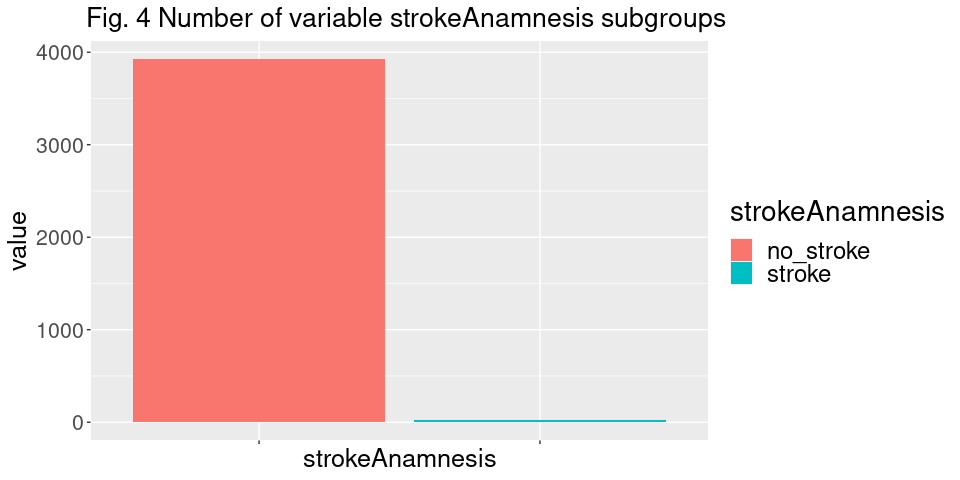
##   
## [[3]]



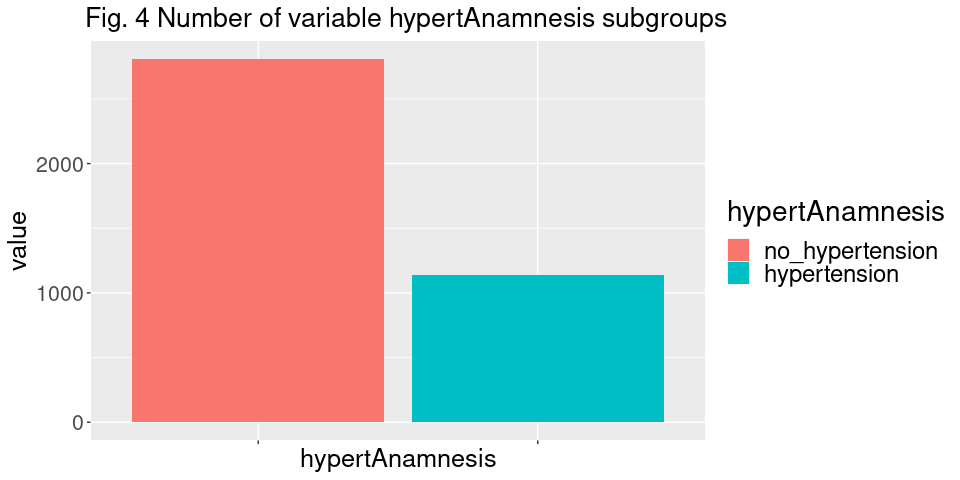
##   
## [[4]]



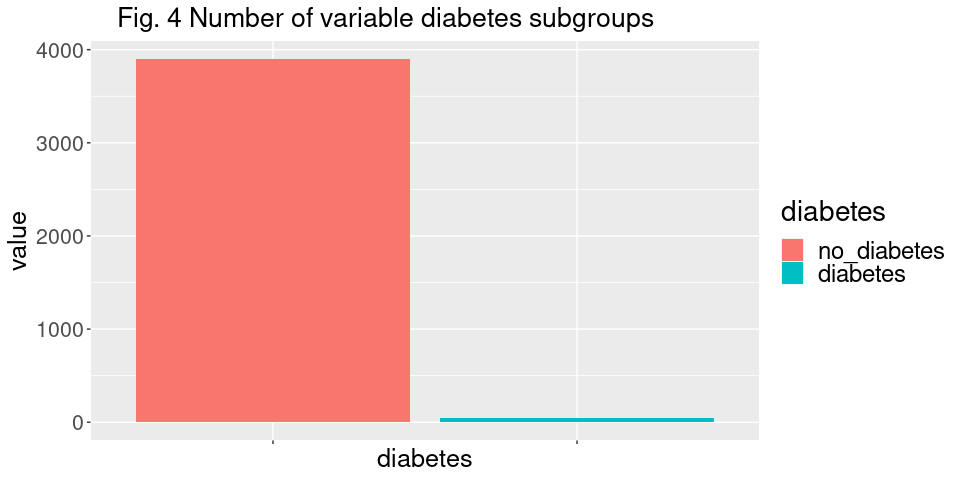
##   
## [[5]]



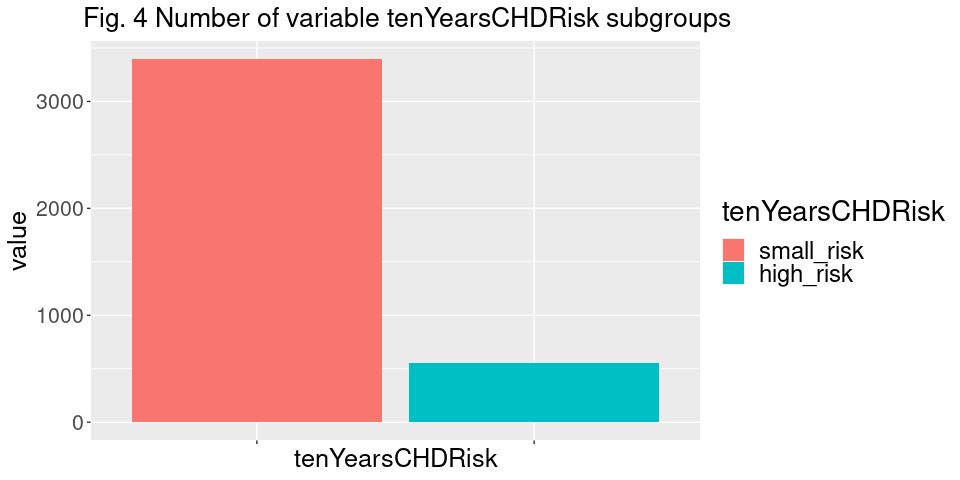
##   
## [[6]]



##   
## [[7]]



##   
## [[8]]



# Статистические оценки

## Проверка на нормальность

1. Оцените каждую переменную на соответствие нормальному распределению с помощью теста Шапиро-Уилка. Какие из переменных являются нормальными и как как вы это поняли?

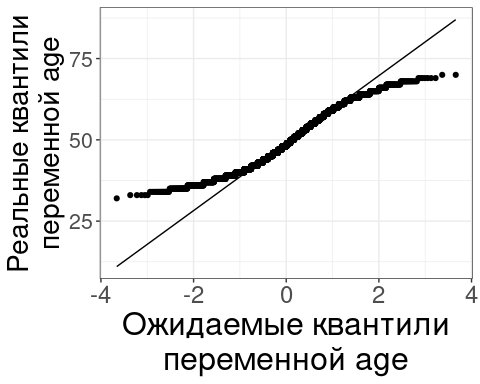
cleaned\_data %>%   
 select(where(is.numeric)) %>%   
 sapply(., function(x) shapiro.test(x))

## age cigarsPerDay   
## statistic 0.9664089 0.7676435   
## p.value 2.04157e-29 1.802036e-59   
## method "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"  
## data.name "x" "x"   
## cholTotal sysBloodPressure   
## statistic 0.9926466 0.9608691   
## p.value 2.280229e-13 2.123379e-31   
## method "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"  
## data.name "x" "x"   
## diaBloodPressure bodyMassIndex   
## statistic 0.990305 0.9907466   
## p.value 8.353604e-16 2.356899e-15   
## method "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"  
## data.name "x" "x"   
## heartRate glucose   
## statistic 0.9828331 0.9397909   
## p.value 1.678202e-21 4.28764e-36   
## method "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"  
## data.name "x" "x"

1. По данным теста Шапиро-Уилка в представленных данных нет нормально распределенных переменных. Это ясно из низких уровней p-value (<0.05).
2. Постройте для каждой количественной переменной QQ-плот. Отличаются ли выводы от теста Шапиро-Уилка? Какой метод вы бы предпочли и почему?

cols <- cleaned\_data %>%   
 select(where(is.numeric)) %>%   
 colnames  
  
qq\_plots <- lapply(cols, function(x) ggplot(cleaned\_data, aes(sample = pull(cleaned\_data[x])))+  
 geom\_qq()+  
 stat\_qq\_line()+  
 labs(x = glue("Ожидаемые квантили\nпеременной {x}"),  
 y = glue("Реальные квантили\nпеременной {x}"))+  
theme\_bw()+  
theme(axis.text.x = element\_text(size=18),  
 axis.text.y = element\_text(size=16),  
 axis.title.x = element\_text(size=24),  
 axis.title.y = element\_text(size=21),  
 plot.title = element\_text(size=25, hjust=0.5)))  
  
qq\_plots

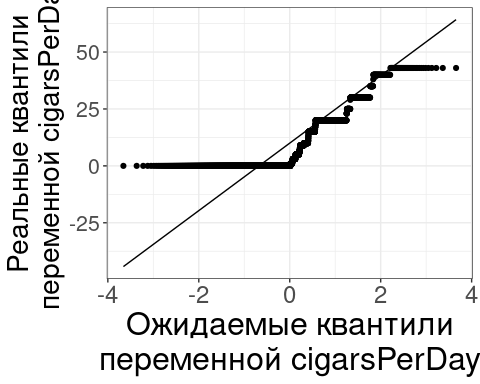
## [[1]]



##   
## [[2]]

## Warning: Removed 22 rows containing non-finite values (`stat\_qq()`).

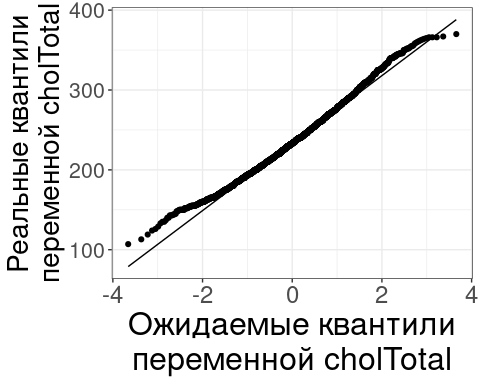
## Warning: Removed 22 rows containing non-finite values (`stat\_qq\_line()`).



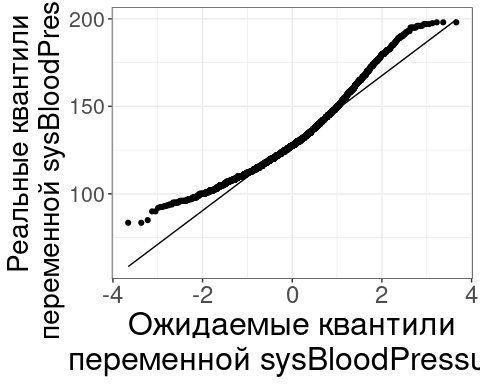
##   
## [[3]]

## Warning: Removed 9 rows containing non-finite values (`stat\_qq()`).

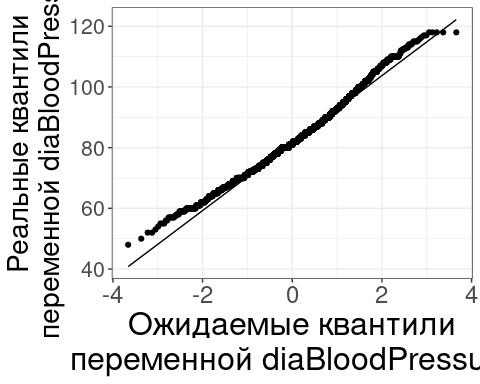
## Warning: Removed 9 rows containing non-finite values (`stat\_qq\_line()`).



##   
## [[4]]



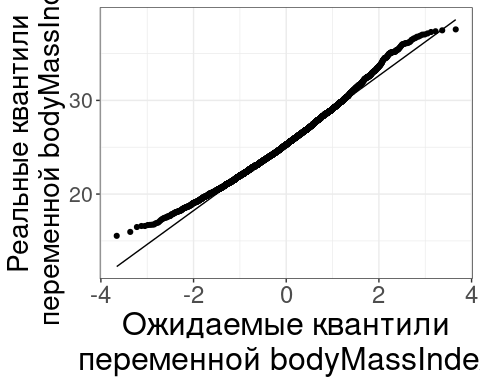
##   
## [[5]]



##   
## [[6]]

## Warning: Removed 12 rows containing non-finite values (`stat\_qq()`).

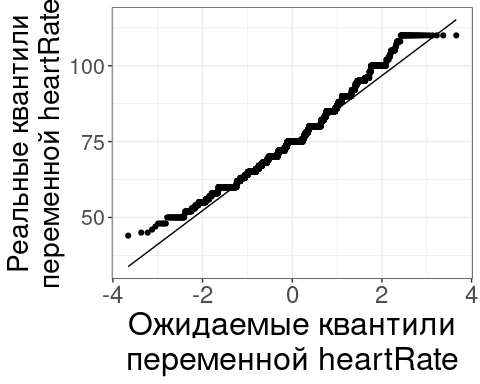
## Warning: Removed 12 rows containing non-finite values (`stat\_qq\_line()`).



##   
## [[7]]

## Warning: Removed 1 rows containing non-finite values (`stat\_qq()`).

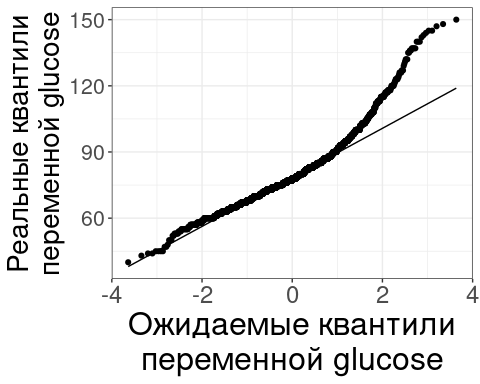
## Warning: Removed 1 rows containing non-finite values (`stat\_qq\_line()`).



##   
## [[8]]

## Warning: Removed 318 rows containing non-finite values (`stat\_qq()`).

## Warning: Removed 318 rows containing non-finite values (`stat\_qq\_line()`).



На выборках с большим количеством единиц наблюдения тест Шапиро-Уилка склонен чаще отклонять нулевую гипотезу о неотличимости тестируемого распределения от нормального (повышается число ошибок первого рода). Поэтому в данном случае я бы из двух подходов предпочел визуальную оценку квантиль-квантильных графиков: на оных для значений общего холестерола, диастолического давления, ИМТ и ЧСС хвосты графиков не очень сильно отклоняются от должных значений. Вместе с тем, визуальная оценка имеет недостатком то, что она, собственно, менее формальна и отчасти зависит от субъективного восприятия, чувства прекрасного и опыта человека, анализирующего ее.

1. Ниже напишите, какие ещё методы проверки на нормальность вы знаете и какие у них есть ограничения.

Тест Колмогорова-Смирнова – еще один формальный подход, сравнивающий данное распределение с неким заранее заданным (в нашем случае – нормальным). На маленьких выборках (условно – меньше 100 единиц наблюдения) склонен реже отвергать нулевую гипотезу (повышенное количество ошибок второго рода).

Построение гистограмм распределения (или функций плотности вероятности) – другой визуальный подход. Можно (и, пожалуй, нужно) комбинировать его с наложением кривой плотности вероятности нормального распределения. Как визуальный метод страдает от повышенной субъективизации оценки. В варианте с гистограммой на восприятие также может влиять такой параметр, как число столбцов.

## Сравнение групп

1. Сравните группы (переменная **TenYearCHD**) по каждой переменной (как количественной, так и категориальной). Для каждой переменной выберите нужный критерий и кратко обоснуйте его выбор в комментариях.
2. Все количественные переменные будут сравниваться t-тестом Стьюдента. Не воспроизведу глубокие математические причины, но бытует мнение, что на больших объемах выборок требование нормального распределения данных становится слабее (может быть это происходит из-за увеличения точности оценки? Но я не уверен). #Дисперсия между группами будет сравниваться при помощи теста Бартлетта. Если дисперсии будут значимо отличаться – используем t-тест с поправкой Уэлча, если не будут – то без него.#
3. Категориальные переменные с двумя градациями и малым (меньше пяти) числом наблюдений хотя бы в одной из ячеек будут оценены точным критерием Фишера. Если же групп будет более двух, а число наблюдений во всех ячейках будет превышать пять, используем критерий хи-квадрат.

# Количественные переменные  
  
num\_cols <- cleaned\_data %>%   
 select(where(is.numeric)) %>%   
 colnames  
  
num\_comp\_list <- lapply(num\_cols, function(x) t.test(pull(cleaned\_data[x]) ~ pull(cleaned\_data["tenYearsCHDRisk"])))  
  
names(num\_comp\_list) <- num\_cols

# Качественные переменные  
  
fac\_names <- cleaned\_data %>%   
 select(where(is.factor) & !tenYearsCHDRisk) %>%   
 colnames  
  
sapply(fac\_names, function(x) length(table(pull(cleaned\_data[x]))))

## sex education currentSmoker bloodPressureDrugs   
## 2 4 2 2   
## strokeAnamnesis hypertAnamnesis diabetes   
## 2 2 2

Оценим число значений в каждой категории переменной education (единственной переменной более чем с двумя градациями).

print(table(subset(cleaned\_data, tenYearsCHDRisk=="high\_risk")$education))

##   
## some\_high\_school high\_school\_GED   
## 273 127   
## some\_college\_vocational\_school college   
## 73 66

print(table(subset(cleaned\_data, tenYearsCHDRisk=="small\_risk")$education))

##   
## some\_high\_school high\_school\_GED   
## 1301 1059   
## some\_college\_vocational\_school college   
## 577 387

Выглядит так, что мы можем использовать хи-квадрат.

chisq.test(cleaned\_data$education, cleaned\_data$tenYearsCHDRisk)

##   
## Pearson's Chi-squared test  
##   
## data: cleaned\_data$education and cleaned\_data$tenYearsCHDRisk  
## X-squared = 29.634, df = 3, p-value = 1.648e-06

fac\_names <- fac\_names[! fac\_names %in% "education"]  
  
fac\_comp\_res <- lapply(fac\_names, function(x)  
 fisher.test(pull(cleaned\_data[x]), pull(cleaned\_data["tenYearsCHDRisk"])))  
  
names(fac\_comp\_res) <- fac\_names  
  
fac\_comp\_res

## $sex  
##   
## Fisher's Exact Test for Count Data  
##   
## data: pull(cleaned\_data[x]) and pull(cleaned\_data["tenYearsCHDRisk"])  
## p-value = 1.153e-09  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 1.459853 2.115580  
## sample estimates:  
## odds ratio   
## 1.756638   
##   
##   
## $currentSmoker  
##   
## Fisher's Exact Test for Count Data  
##   
## data: pull(cleaned\_data[x]) and pull(cleaned\_data["tenYearsCHDRisk"])  
## p-value = 0.06635  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 0.9884218 1.4293897  
## sample estimates:  
## odds ratio   
## 1.188359   
##   
##   
## $bloodPressureDrugs  
##   
## Fisher's Exact Test for Count Data  
##   
## data: pull(cleaned\_data[x]) and pull(cleaned\_data["tenYearsCHDRisk"])  
## p-value = 0.001524  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 1.330380 3.565235  
## sample estimates:  
## odds ratio   
## 2.210361   
##   
##   
## $strokeAnamnesis  
##   
## Fisher's Exact Test for Count Data  
##   
## data: pull(cleaned\_data[x]) and pull(cleaned\_data["tenYearsCHDRisk"])  
## p-value = 0.0003472  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 2.003398 13.213822  
## sample estimates:  
## odds ratio   
## 5.204175   
##   
##   
## $hypertAnamnesis  
##   
## Fisher's Exact Test for Count Data  
##   
## data: pull(cleaned\_data[x]) and pull(cleaned\_data["tenYearsCHDRisk"])  
## p-value < 2.2e-16  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 2.052132 2.990950  
## sample estimates:  
## odds ratio   
## 2.477903   
##   
##   
## $diabetes  
##   
## Fisher's Exact Test for Count Data  
##   
## data: pull(cleaned\_data[x]) and pull(cleaned\_data["tenYearsCHDRisk"])  
## p-value = 0.09547  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 0.8237568 3.6166613  
## sample estimates:  
## odds ratio   
## 1.798048

# Далее идут **необязательные** дополнительные задания, которые могут принести вам дополнительные баллы в том числе в случае ошибок в предыдущих

## Корреляционный анализ

1. Создайте корреляционную матрицу с визуализацией и поправкой на множественные сравнения. Объясните, когда лучше использовать корреляционные матрицы и в чём минусы и плюсы корреляционных исследований.
2. Уточню, что использовался коэффициент корреляции Спирмена, а не Пирсона, поскольку ранее было показано, что для многих количественных переменных распределение статзначимо отличается от нормального. Будем консервативными, и в качестве поправки на множественное сравнение используем поправку Бонферрони (я не обнаружил, как быстро задать факт поправки в визуализации, поэтому “поднял” уровень p-значения вручную).

Корреляционная матрица – удобный способ, например, найти коллинеарные предикторы и не включать их в регрессионный анализ. Также это хороший подход, чтобы “посмотреть на данные” количественного типа и подумать, что с ними делать дальше. Всякого рода огромные таблицы со спирменовскими (чаще всего) корреляциями используются для верификации моделей в контексте машинного обучения, но тут я разбираюсь плохо, поэтому, весьма возможно, что в текущем предложении сказал какую-то глупость.

Проблема в том, что наличие корреляции часто воспринимают неправильно – как причинно-следственную, а не статистическую связь (в этой связи уместно вспомнить анекдоты про аистов и рождаемость в Дании после Второй Мировой войны, а также про количество самоубийств в США и число фильмов, в которых Николас Кейдж снялся за год).

Иная проблема в том, что корреляции часто могут искажаться конфаундерами, а при чрезмерно рьяной погоне за оными – коллайдерами.

correlations <- corr.test(cleaned\_data %>% select(where(is.numeric)), method = "spearman", adjust = "bonferroni")  
cormatrix <- ggcorrplot(correlations$r, type = "lower", outline.col = "white", lab=TRUE, method = "circle", p.mat = correlations$p, sig.level = 0.05/(sum(1:length(cleaned\_data %>% select(where(is.numeric)) %>% colnames))))  
cormatrix



## Моделирование (недоделано, в планах завершить к жесткому дедлайну)

1. Постройте регрессионную модель для переменной **TenYearCHD**. Опишите процесс построения
2. Сначала уберем все NA в данных, и просто возьмем и положим в модель логистической регрессии (потому что у нас бинарный исход) все факторы.

# cleaned\_data\_nna <- cleaned\_data %>% na.omit()  
# model <- glm(tenYearsCHDRisk ~., family=binomial(link='logit'),data=cleaned\_data\_nna)  
#   
# summary(model)

Если применять логику обычных линейных моделей, нам нужно исключить коллинеарные факторы. У нас таковыми являются систолическое и диастолическое АД. Удалим диастолическое АД (оно не является значимым предиктором).