Основы статистики

Александр Нехаев

Contents

Chapter 1

Введение

1.1 Понятие генеральной совокупности и выборки, репрезентативность выборки

1.1.1 Понятие выборки и генеральной совокупности

Определение 1.1.1 *Генеральная совокупность – множество всех объеков относительно которой мы хотим делать выводы в рамках исследования некоторой проблемы.*

Определение 1.1.2 *Выборка – часть генеральной совокупности используемой в реальном исследовании.*

Определение 1.1.3 *Репрезентативность выборки – применимость выводов по выборке к генеральной совокупности.*

1.1.2 Выборка

1.1.2.1 Простая случайная выборка (simple random sample)

Случайный выбор элементов из генеральной совокупности

1.1.2.2 Стратифицированная выборка (stratified sample)

Разбиение генеральной совокупности на несколько групп с явно различными свойствами. Затем случайной выборкой берем элементы из каждой группы.

1.1.2.3 Групповая выборка (cluster sample)

Так же разбиваем совокупность на кластеры, которые схожи по свойствам. Затем выбираем несколько кластеров, затем из кластеров выбираем случайные элементы.

1.2 Типы переменных. Количественные и номинативные переменные

1.2.1 Типы переменных

Все переменные характеризующие генеральные совокупности можно разделить на **количественные** и **номинативные**.

1.2.1.1 Количественные переменные

- Непрерывные
- Дискретные

Например: Непрерывная - рост человека из выборки на интервале от 160 до 190 см. Дискретная - количество детей в семье.

1.2.1.2 Номинативные

Используются для разделения элементов выборки на какие-то группы.

1.2.1.3 Ранговые переменные

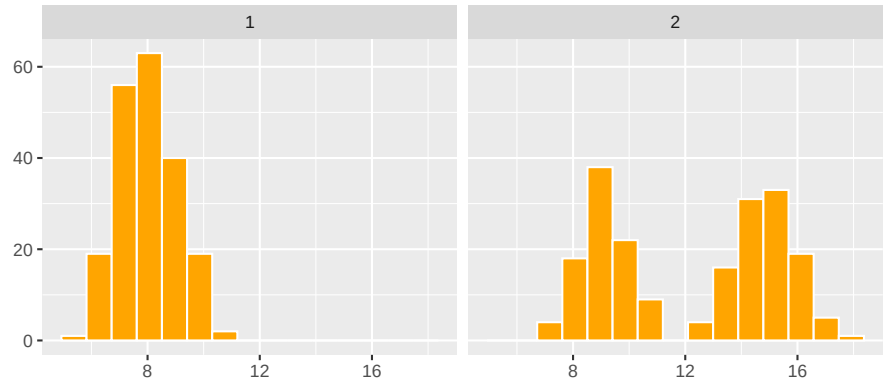
Переменная к которой можно применять только операцию сравнения.

1.3 Меры центральной тенденции

1.3.1 Понятие описательной статистики

Гистограмма частот - первое описание формы распределения.

```
bind_rows(
    tibble(x = rnorm(200, mean = 8, sd = 1)),
    tibble(
        x = r(UnivarMixingDistribution(
            Norm(mean = 9, sd = 1),
            Norm(mean = 15, sd = 1)
        ))(200)
    ),
    .id = "id"
) |>
    ggplot(aes(x)) +
    facet_grid(cols = vars(id)) +
    geom_histogram(
        bins = 15,
        color = "white",
        fill = "orange"
    ) +
    labs(x = "", y = "")
```



1.3.1.1 Мера центральной тенденции

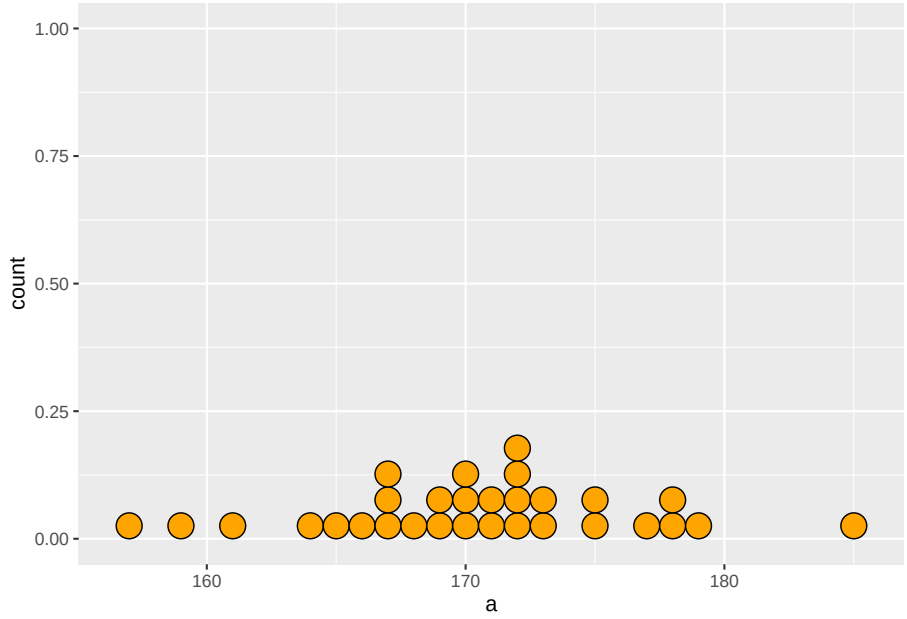
Отвечает на вопрос насколько высокие значения принимает переменная

1.3.2 Мода

Определение 1.3.1 *Мода – значение измеряемого признака, которая встречается максимально часто*

Пример: Пусть есть выборка:

```
a <- c(
    185, 175, 170, 169, 171, 172, 175, 157, 170, 172, 167,
    173, 168, 167, 166, 167, 169, 172, 177, 178, 165, 161,
    179, 159, 164, 178, 172, 170, 173, 171
)
a_dot_plot <- tibble(a = a) |>
    ggplot(aes(a)) +
    geom_dotplot(binwidth = 1, fill = "orange")
a_dot_plot
```




```
st.mode <- function(x) {
    u <- unique(x)
    tab <- tabulate(match(x, u))
    u[tab == max(tab)]
}
st.mode(a)
```

```
## [1] 172
```

1.3.3 Медиана

Определение 1.3.2 *Медиана – значение признака, которое делит упорядоченное множество данных пополам.*

```
b <- c(157, 159, 161, 164, 165, 166, 167, 167, 167)
length(b)
```

```
## [1] 9
```

```
median(b)
```

```
## [1] 165
```

В случае если у нас нечетное количество элементов – все просто. Если нечетное, то берем среднее значение двух значений между которыми находится середина.

```
median(a)
```

```
## [1] 170.5
```

1.3.4 Среднее значение

Определение 1.3.3 *Среднее значение – сумма всех значений измеренного признака, деленая на количество измеренных значений.*

```
mean(a)
```

```
## [1] 170.4
```

1.3.5 Выбор меры центральной тенденции

Смотрим на все значения, которые мы получили:

```
tibble(x = a) |>
    summarise(
        Mode = st.mode(a),
        Median = median(a),
        Mean = mean(a)
    ) |>
    kable()
```

Mode	Median	Mean
172	170.5	170.4

Если распределение симметрично, унимодально и не имеет заметных выбросов, то все тенденции дадут примерно одно значение. Если оно симметрично, с выбросами или мультимодально, тогда лучше использовать моду или медиану.

1.3.6 Свойства среднего

$$M_{x+c} = M_x + c \tag{1.1}$$

$$M_{x+c} = M_x * c \tag{1.2}$$

$$\sum (x_i - M_x) = 0 \tag{1.3}$$

Проверка:

```
cn <- round(rnorm(200))
c(
    mean(cn + 2) == mean(cn) + 2,
    mean(cn * 2) == mean(cn) * 2,
    round(sum(cn - mean(cn))) == 0
)
```

```
## [1] TRUE TRUE TRUE
```

1.4 Меры изменчивости

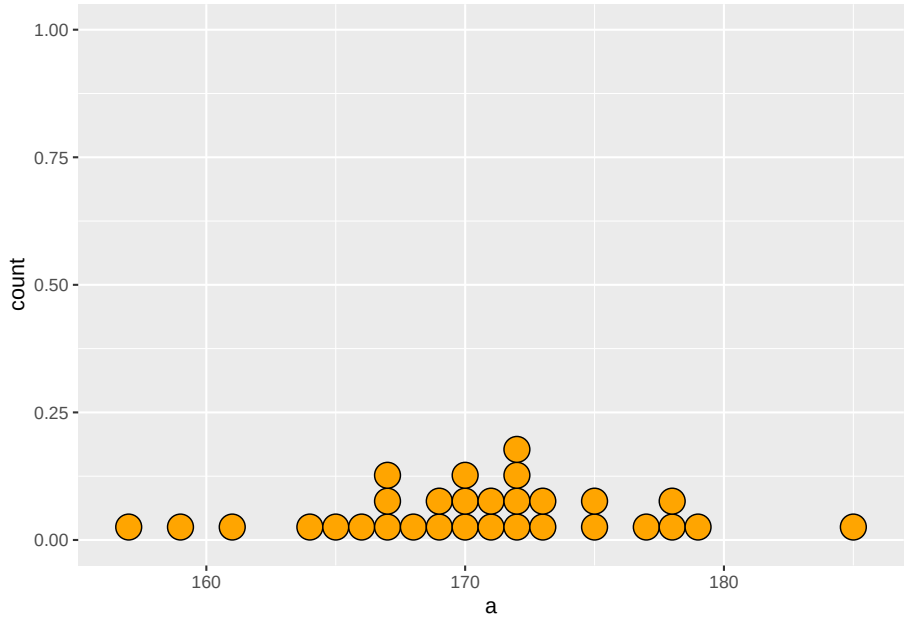
1.4.1 Понятие меры изменчивости данных

Некоторые распределения имеют значительные отличия даже несмотря на близкие значения среднего, медианы и моды. Для описания из различий используются меры изменчивости.

1.4.2 Размах

Определение 1.4.1 *Размах – разность максимального и миниамального значений*

```
a_dot_plot
```




```
max(a) - min(a)
```

```
## [1] 28
```

Изменение краевых величин будет сильно влиять на эту меру.

1.4.3 Дисперсия, стандартное отклонение

Дисперсия – средний квадрат отклонений индивидуальных значений признака от их средней величины.

Среднее отклонение от среднего вообще говоря имеет вид:

$$\frac{\sum (x_i - \overline{x})}{n}$$

Но как мы знаем из свойств среднего, числитель тогда будет равен 0. Исключаем отрицательные значения через возведение в квадрат:

$$\frac{\sum (x_i - \overline{x})^2}{n} \tag{1.4}$$

Это и называется дисперсией.

```r
var(a)
```

```
## [1] 36.04138
```

Однако мы взяли квадрат, что сильно влияет на результат (можно сказать, что поменялась размерность). Поэтому более точной величиной будет корень из нее.

Стандартное отклонение - корень из дисперсии.

```r
sd(a)
```

```
## [1] 6.003447
```

Тут стоит отметить, что в литературе для обозначения стандартного отклонения для всей генеральной совокупности используется символ σ. Для стандартного отклонения используется sd.

Еще момент. Если мы считаем дисперсию для всей генеральной совокупности, то мы смело используем формулу (0.1). Однако если мы берем дисперсию для совокупности, то в знаменателе используем $n-1$. Это связано со степенями свободы.

1.4.3.1 Пример

```r
example <- c(1, 2, 2, 3, 4, 4, 5)
mean(example)
```

```
## [1] 3
```

```r
(example - mean(example))^2
```

```
## [1] 4 1 1 0 1 1 4
```

```r
sum((example - mean(example))^2) / (length(example) - 1)
```

```
## [1] 2
```

```r
sqrt(sum((example - mean(example))^2) / (length(example) - 1))
```

```
## [1] 1.414214
```

Используя готовые функции:

```
tibble(x = example) |>
    summarise(
        "Variance" = var(x),
        "Standard deviation" = sd(x)
    ) |>
    kable()
```

Variance	Standard deviation
2	1.414214

1.4.4 Свойства дисперсии и стандартного отклонения

$$D_{x+c} = D_x \tag{1.5}$$

$$sd_{x+c} = sd_x \tag{1.6}$$

$$D_{x*c} = D_x * c^2 \tag{1.7}$$

$$sd_{x*c} = sd_x + c \tag{1.8}$$

1.5 Квартили распределения и график box-plot

1.5.1 Квантили распределения

Квантили - это такие значения признака, которые делят упорядоченные данные на некоторое число равных частей Примером квартиля является медиана, которую мы уже рассматривали, однако в статистике так же часто используются квартили распределения - 3 точки, которые делят распределение на 4 равных части. ### Квартили *Квартили - три точки (значения признака), которые делят* ***упорядоченное*** *множество данных на 4 равные части.*

```
sort(a)
```

```
##  [1] 157 159 161 164 165 166 167 167 167 168 169 169 170 170
## [15] 170 171 171 172 172 172 172 173 173 175 175 177 178 178
## [29] 179 185
```

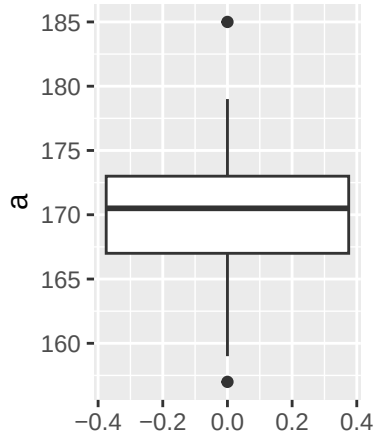
```
quantile(a) |> kable()
```

	x
0%	157.0
25%	167.0
50%	170.5
75%	173.0
100%	185.0

1.5.2 Box-plot

Верхняя и нижняя границы коробки отражают положение 1го и 3го квартилей. Линия внутри коробки - 2й квартиль (медиана). Положения усов - последние значения в пределе 1.5 межквартильных размахов от границ коробки. Точки - выпадающие из них значения.

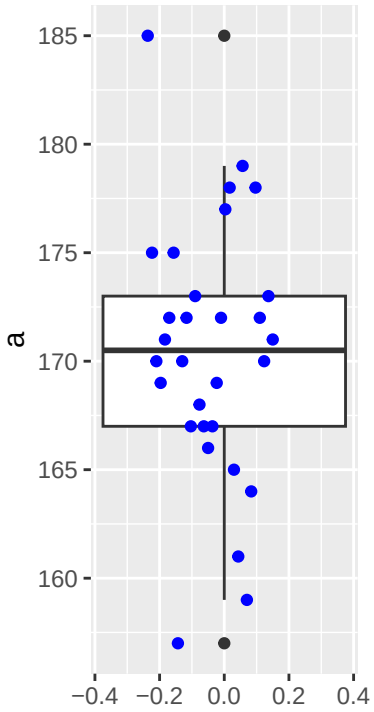
```
tibble(a = a) |> ggplot(aes(y = a)) +
    geom_boxplot()
```



Теперь нанесем точки на график с box-plot:

```
tibble(a = a) |>
    mutate(
        n = row_number(),
        x = (n / length(a)) * 0.4 - 0.25
    ) |>
    ggplot(aes(y = a)) +
```

```
    geom_boxplot() +
    geom_point(aes(x = x), color = "blue") +
    labs(x = "")
```



Box plot не столь информативен, сколько гистограмма, однако он помогает при сравнении двух распределений.

1.6 Нормальное распределение

1.6.1 Понятие нормального распределения

Характеристики:

- Унимодально
- Симметрично
- Отклонение наблюдений от среднего подчиняется определенному вероятностному закону:

- В диапазоне от медианы до среднеквадратичного отклонения будет находиться примерно 34.1% всех значений.
- В диапазоне от одного до двух среднеквадратичных отклонений будет находиться примерно 13.6%.
- Вероятность встретить значение, превосходящее 3 стандартных отклонения весьма маловероятна (там около 0.1% значений)

1.6.2 Стандартизация

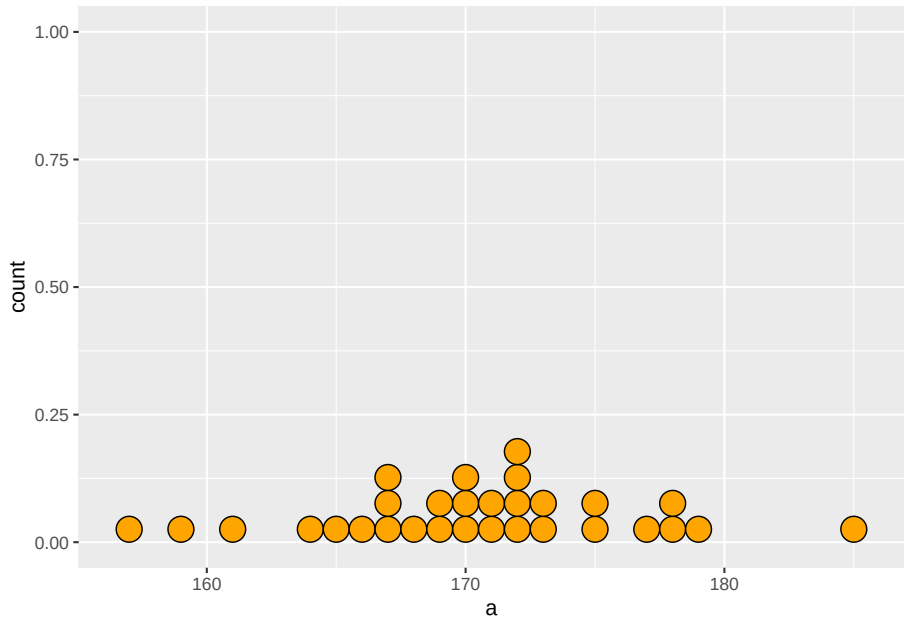
Стандартизация или z-преобразование – преобразование полученных данных в стандартную Z-шкалу (Z-scores) со средним $M_z = 0$ *и* $D_z = 1$.

Для этого:

$$Z_i = \frac{x_i - \overline{x}}{\sigma_x}$$

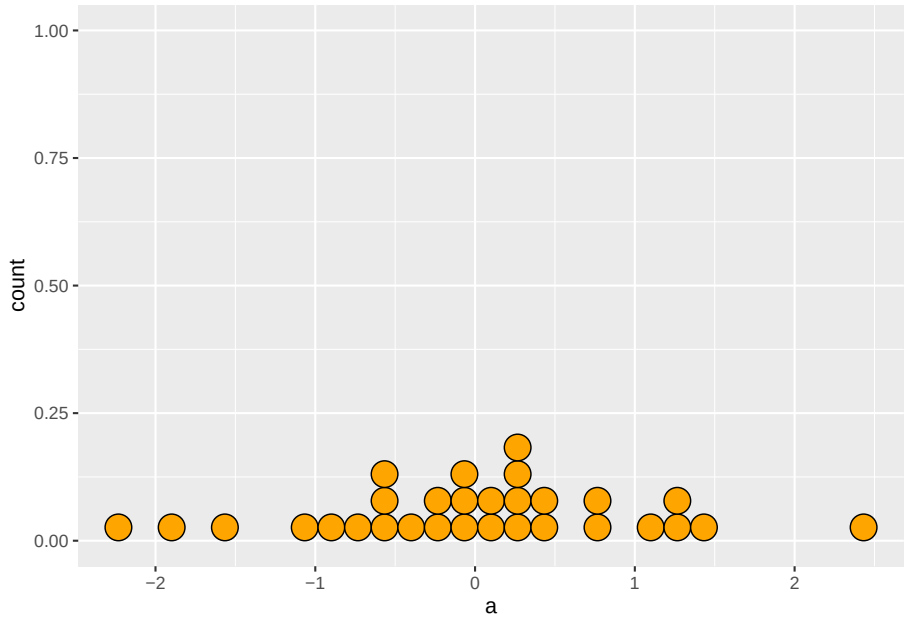
Видим, что форма распределения не меняется, меняются только значения:

```
tibble(a = a) |>
    ggplot(aes(x = a)) +
    geom_dotplot(binwidth = 1, fill = "orange")
```




```
tibble(a = base::scale(a)) |>
    ggplot(aes(x = a)) +
```

```
    geom_dotplot(
        binwidth = 1 / length(a),
        dotsize = 5,
        fill = "orange"
    )
```



1.6.3 Правила двух и трех сигм, использование стандартизации

Ранее уже говорили, что:

- $M_x \pm \sigma \approx 68\%$ наблюдений
- $M_x \pm 2\sigma \approx 95\%$ наблюдений
- $M_x \pm 3\sigma \approx 100\%$ наблюдений

z-преобразование позволяет ответить на вопрос какой процент наблюдений лежит в любом заданном диапазоне.

Пример: Мы хотим узнать какой процент значений превышает значение 154 если среднее значение составляет 150, а стандартное отклонение равно 8.

Находим z-значение для заданного значения:

```
z <- (154 - 150) / 8
print(z)
```

```
## [1] 0.5
```

```
pnorm(-z, mean = 0, sd = 1)
```

```
## [1] 0.3085375
```

1.7 Центральная предельная теорема

Допустим, что некоторый признак распределен нормально в генеральной совокупности имеет среднее значение и стандартное отклонение:

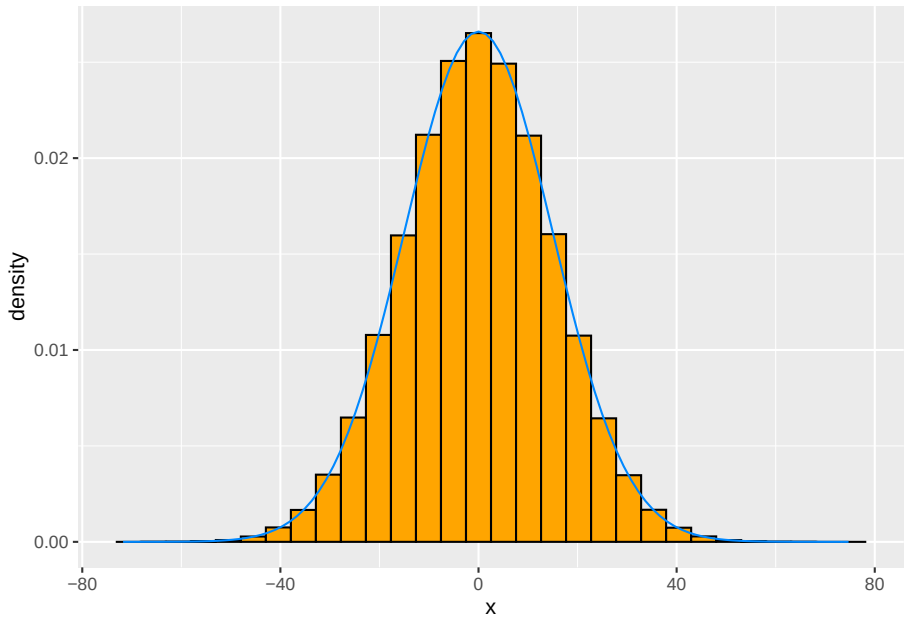
```
mu <- 0
sigma <- 15
```

Обозначаем эту совокупность как

```
dist <- rnorm(175 * 5000, mean = mu, sd = sigma)
```

и строим её:

```
tibble(x = dist) |>
    ggplot(aes(x = x)) +
    geom_histogram(
        bins = 30,
        aes(y = after_stat(density)),
        color = "black", fill = "orange"
    ) +
    stat_function(
        fun = dnorm,
        args = list(mean = mu, sd = sigma),
        color = "#0088ff"
    )
```



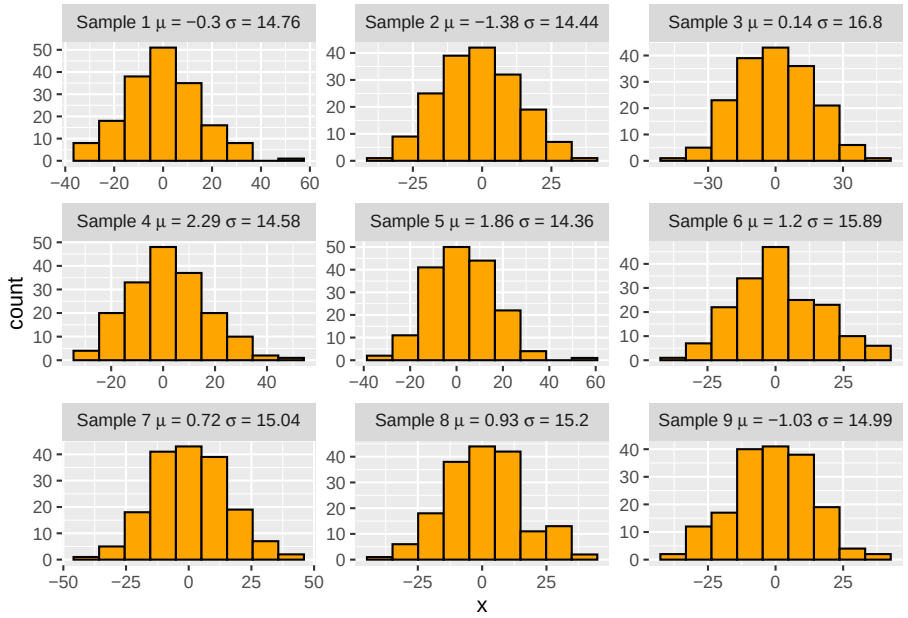
Будем многократно извлекать из совокупности выборки по 175 значений каждая и замерять в них среднее значение и стандартное отклонение (отображены первые 9 выборок):

```
sample_size <- 175
samples <- bind_rows(map(1:5000, \(id) tibble(x = dist) |>
    slice_sample(n = sample_size)), .id = "bin") |>
    mutate(bin = as.numeric(bin))
plt_names <- samples |>
    filter(bin <= 9) |>
    group_by(bin) |>
    summarise(mu = round(mean(x), 2), sd = round(sd(x), 2))

means <- plt_names$mu
names(means) <- plt_names$bin
sds <- plt_names$sd
names(sds) <- plt_names$bin

samples |>
    filter(bin <= 9) |>
    ggplot(aes(x = x)) +
    facet_wrap(vars(bin),
        scales = "free",
        labeller = label_bquote(
            "Sample " * .(bin) * " " * mu * " = " *
```

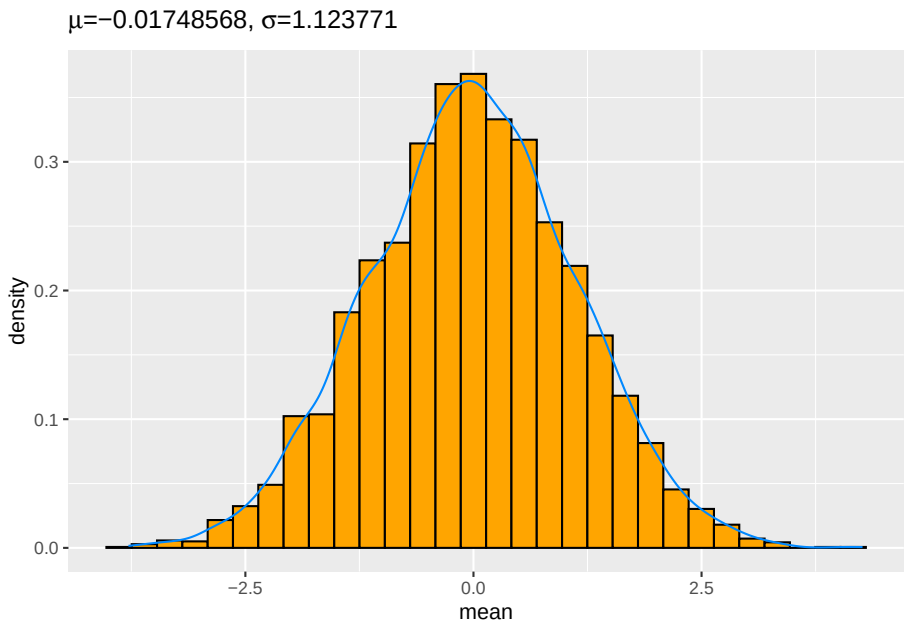
```
                .(means[bin]) * " " * sigma *
                " = " * .(sds[bin])
        )
    ) +
    geom_histogram(
        bins = 9,
        color = "black",
        fill = "orange"
    )
```



Теперь возьмем рассчитанные значения среднего и стандартного отклонения для всех выборок и отобразим их на графике:

```
samples_data <- samples |>
    group_by(bin) |>
    summarise(
        mean = mean(x),
        sd = sd(x)
    )
samples_data |> ggplot(aes(x = mean)) +
    geom_histogram(
        aes(y = after_stat(density)),
        bins = 30, color = "black", fill = "orange"
    ) +
    geom_density(kernel = "gaussian", color = "#0088ff") +
```

```
    labs(
        title = bquote(paste(
            mu, "=", .(mean(samples_data$mean)),
            ", ", sigma, "=", .(sd(samples_data$mean))
        ))
    )
```



Значение σ на этом графике называется **стандартной ошибкой среднего** и показывает на сколько в среднем значение выборочного среднего отклоняется от среднего генеральной совокупности. С ростом количества элементов в выборке стандартная ошибка среднего будет уменьшаться.

Таким образом формулируем центральную предельную теорему.

Теорема 1.7.1 *Предположим исследуемый признак имеет нормальное распределение в генеральной совокупности с некоторым средним значением и стандартным отклонением и мы многократно извлекаем выборки равные по объему n и в каждой выборке рассчитываем среднее значение после чего строим распределение средних значений. Такое распределение будет являться нормальным со средним, совпадающем со среднем генеральной совокупности и со стандартным отклонением, называемым стандартной ошибкой среднего и рассчитываемым по формуле:*

$$se = \frac{\sigma}{\sqrt{n}} \tag{1.9}$$

Замечание: на самом деле исследуемый признак может иметь любое распределение и средние выборок так же буду распределены нормально.

Чем больше элементов в выборке, тем ближе среднее значение в каждой выборке к среднему значению генеральной совокупности и соответственно тем меньше будет стандартная ошибка среднего. Так же есть правило, что если число элементов в выборке больше 30 и эта выборка репрезентативная, то формулу из теоремы можно преобразовать до вида:

$$se = \frac{sd(x)}{\sqrt{n}}. \tag{1.10}$$

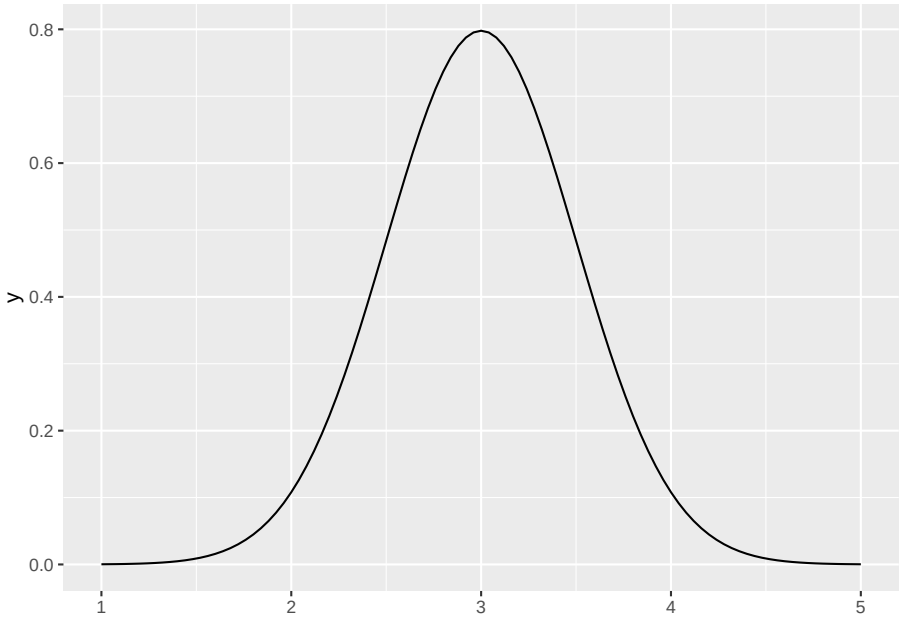
Пусть мы извлекли из совокупности всего одну выборку в 100 элементов. Стандартное отклонение 5 и среднее значение 3. На основе этих данных мы можем предположить, как вели бы себя все выборки этой совокупности:

```
5 / sqrt(100)
```

```
## [1] 0.5
```

Соответственно распределение средних значений выборок имело бы вид:

```
ggplot() +
    xlim(1, 5) +
    stat_function(
        fun = dnorm,
        args = list(mean = 3, sd = 5 / sqrt(100))
    )
```



Утверждается, что данное распределение будет получатся во всех выборках.

1.8 Идея статистического вывода, p-уровень значимости

1.8.1 Статистическая проверка гипотез

Как правило нас все таки интересуют гипотезы, а не конкретные значения. Рассмотрим пример:

Предположим, что на выздоровление при некотором заболевании в среднем требуется $M = 20$ дней, но мы разработали новый препарат и хотим проверить может ли он сократить этот срок. Мы взяли 64 пациента и опробовали на них новый метод лечения. Оказалось, что средняя скорость выздоровления сократилась до $x = 18.5$ дней при среднем стандартном отклонении $sd = 4$. Какой вывод можно сделать из этих данных?

С одной стороны судя по значениям, мы действительно сократили срок выздоровления. С другой стороны, такой результат мог быть получен случайно и без препарата. Введем несколько важных понятий.

В этом исследовании будут конкурировать 2 гипотезы:

- H_0 – никакого реального воздействия препарат не оказывает и

на самом деле среднее значение генеральной совокупности тех пациентов, которые получили препарат на самом деле не отличается от генеральной совокупности всех больных, $M_{\text{НП}} = 20$.
- H_1 – препарат влияет и среднее значение скорости восстановления генеральной совокупности всех пациентов, использующих препарат отличается, $M_{\text{НП}} \neq 20$.

Пусть на самом деле верна первая гипотеза. Тогда согласно ЦПТ если бы мы многократно повторяли исследование, то выборочные средние распределились бы нормальным образом вокруг среднего генеральной совокупности с стандартной ошибкой $se = \frac{sd}{\sqrt{n}} = \frac{4}{\sqrt{64}} = 0.5$. Теперь ответим на вопрос «насколько далеко наше выборочное среднее отклонилось от предполагаемого среднего значения генеральной совокупности в единицах стандартного отклонения?» Для этого сделаем z-преобразование:

$$z = \frac{\overline{x} - M}{se} = \frac{18.5 - 20}{0.5} = -3$$

Это означает, что если бы в генеральной совокупности среднее значение на самом деле равнялось бы 20, то выборочно среднее отклонилось бы от среднего генеральной совокупности на -3σ влево.

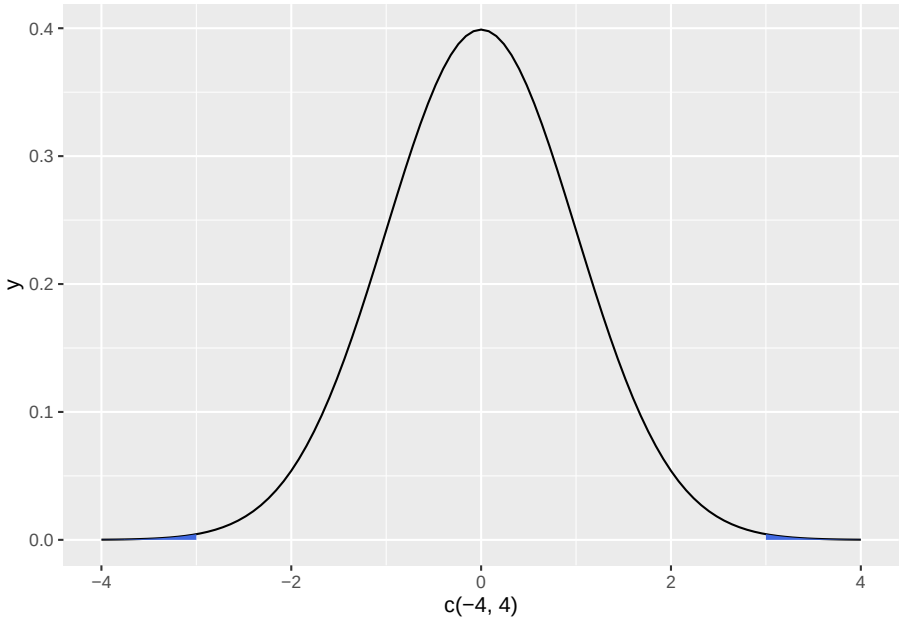
1.8.2 Идея статистического вывода

Теперь воспользуемся свойствами нормального распределения чтобы рассчитать вероятность такого или еще более сильно выраженного отклонения от среднего значения:

```
pnorm(-3) + pnorm(3, lower.tail = FALSE)
```

```
## [1] 0.002699796
```

```
ggplot() +
    xlim(-4, 4) +
    stat_function(fun = dnorm) +
    geom_area(
        stat = "function", fun = dnorm,
        fill = "royalblue", xlim = c(-4, -3), aes(c(-4, 4))
    ) +
    geom_area(
        stat = "function", fun = dnorm,
        fill = "royalblue", xlim = c(3, 4), aes(c(-4, 4))
    )
```



Итак, на первом этапе мы предположили, что на самом деле верна нулевая гипотеза. Если это так, то все выборочные средние распределились бы вокруг среднего генеральной совокупности, которое, как мы предполагаем, равняется 20. Однако в нашем эксперименте выборочное среднее оказалось равно 18.5. Зная стандартную ошибку среднего мы смогли рассчитать вероятность получить такое или еще более сильно выраженное отклонение причем как в правую, так и в левую сторону. Оказалось, что такая вероятность ≈ 0.003.

Таким образом основная **идея статистического вывода** заключается в следующем: мы допускам, что верна нулевая гипотеза и на самом деле никаких различий у нас нет. Затем мы считаем вероятность того, что мы получили такие или еще более сильно выраженные различия абсолютно случайно. Это значение в статистике называется p-уровень значимости и именно при помощи этого показателя мы выясним какую гипотезу считать более состоятельной. Чем меньше p-уровень, тем больше оснований отклонить нулевую гипотезу. Считается, что если $p < 0.05$, то можно смело принимать альтернативную гипотезу. Однако если p-уровень больше этого порога, у нас недостаточно оснований отклонить эту гипотезу.

1.8.3 p-уровень значимости и его интерпретация

Получаем, что у нас достаточно оснований для отклонения нулевой гипотезы. Вопрос - зачем рассчитывать значение отклонения в принципиально другом направлении? Ведь если мы проверили гипотезу о

том, что препарат ускорит скорость выздоровления, то зачем учитывать вероятность, что он её понизит? Вероятность это площадь под кривой. Если рассматривать только одно направление, то вероятность события будет меньше. Тем не менее принято учитывать оба конца распределения. Реально мы не знаем в какую сторону мы получим отклонение от среднего и от такого развития событий никто не застрахован. Иногда действительно используется односторонний p-критерий, обычно если отклонение в другую сторону невозможно.

В реальности p-уровень означает, что если верна нулевая гипотеза, то вероятность получить такие или еще более выраженные различия будет равна p-уровню. Он ничего не говорит о силе эффекта. Мы можем получить в среднем сокращение времени болезни на неделю, но при этом не значимое с точки зрения статистики.

Что делать, если уровень значимости оказался больше 0.05? Вывод простой - у нас недостаточно оснований для отклонения нулевой гипотезы. Сам по себе p-уровень ничего не говорит ни о правильности, ни о ценности получаемых результатов.

Основная идея статистической проверки гипотезы подразумевает, что мы иногда будем совершать статистические ошибки 1го и 2го рода.

Определение 1.8.1 ***Ошибка 1-го рода*** *подразумевает, что мы отклонили первую гипотезу, хотя на самом деле она верна.*

Определение 1.8.2 ***Ошибка 2-го рода*** *подразумевает, что мы не отклонили нулевую гипотезу, хотя на самом деле она не верна.*

Chapter 2

Сравнение средних

2.1 T-распределение

2.1.1 Нормальное распределение и ограниченность количества наблюдений

С выборками в которых большое число элементов все понятно. Теперь рассмотрим ситуацию, когда число элементов достаточно мало (<30). Особенность такого случая заключается в том, что нарушается предположение о том, что во-первых, стандартное отклонение среднего уже не такое хорошее, а во-вторых нарушается предположение о том, что все выборочные средние будут вести себя в соответствии с нормальным распределением.

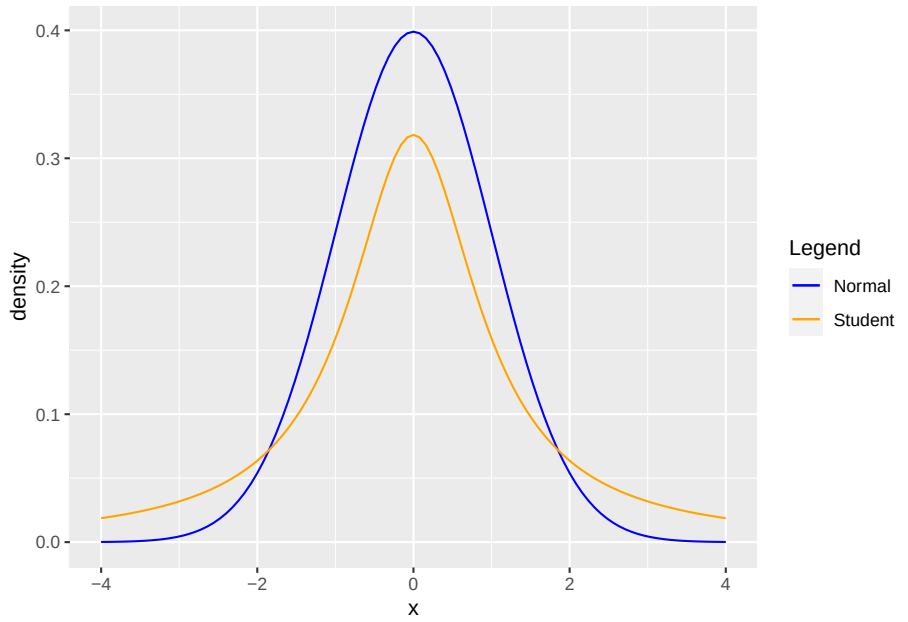
2.1.2 Распределение Стьюдента (t-распределение)

По этим причинам, если число наблюдений невелико и стандартное отклонение неизвестно, то используется распределение Стьюдента (t-распределение).

Распределение Стьюдента унимодально и симметрично, но наблюдения с большой вероятностью попадают за пределы $\pm 2\sigma$ от M.

```
ggplot() +
    xlim(-4, 4) +
    stat_function(fun = dnorm, aes(color = "Normal")) +
    stat_function(
        fun = dt, args = list(df = 1),
        aes(color = "Student")
```

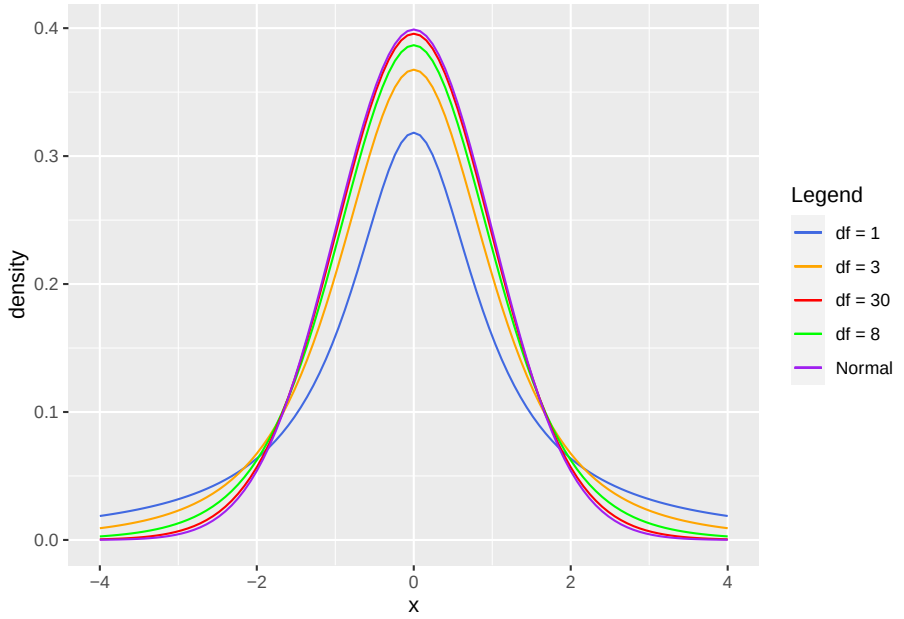
```
    ) +
    labs(
        x = "x",
        y = "density",
        color = "Legend"
    ) +
    scale_color_manual(values = c(
        "Normal" = "blue",
        "Student" = "orange"
    ))
```



Форма распределения определяется числом степеней свободы ($df = n - 1$), где n - число наблюдений в выборке. С увеличением числа df распределение стремится к нормальном.

```
t_student_demo_plot <- ggplot() +
    xlim(-4, 4) +
    stat_function(
        fun = dt, args = list(df = 1),
        aes(color = "df = 1")
    ) +
    stat_function(
        fun = dt, args = list(df = 3),
        aes(color = "df = 3")
```

```
    ) +
    stat_function(
        fun = dt, args = list(df = 8),
        aes(color = "df = 8")
    ) +
    stat_function(
        fun = dt, args = list(df = 30),
        aes(color = "df = 30")
    ) +
    stat_function(fun = dnorm, aes(color = "Normal")) +
    labs(x = "x", y = "density", color = "Legend") +
    scale_color_manual(values = c(
        "df = 1" = "royalblue",
        "df = 3" = "orange",
        "df = 8" = "green",
        "df = 30" = "red",
        "Normal" = "purple"
    ))
t_student_demo_plot
```



Рассмотрим пример. Пусть в генеральной совокупности среднее значение $\mu = 10$. На выборке получили среднее равное $x = 10.8$ со стандартным отклонением $sd = 2$ при числе испытаний $N = 25$.

Если пользоваться стандартной формулой описанной ранее, то мы бы

сказали, что в соответствии с ЦПТ все выборочные средние распределились бы нормально вокруг среднего генеральной совокупности и стандартная ошибка среднего была бы:

```
2 / sqrt(25)
```

```
## [1] 0.4
```

Теперь мы хотим посмотреть насколько наше выборочное среднее отклонилось от среднего генеральной совокупности. Тогда мы сможем найти вероятность получить такое или еще более выраженное отклонение. Для этого ищем соответствующее z-значение:

```
(10.8 - 10) / 0.4
```

```
## [1] 2
```

То есть отклонение составляет 2 стандартных отклонения.

Теперь чуть больше поговорим о том, почему t-распределение необходимо на небольшом объеме выборки.

2.1.2.1 Про необходимость t-критерия

Мы знаем, что если некоторый признак в генеральной совокупности распределен нормально (или согласно какому-либо другому распределению) со средним μ и стандартным отклонением σ и мы будем многократно извлекать выборки одинакового размера n, и для каждой выборки рассчитывать, как далеко выборочное среднее $\overline{X}$ отклонилось от среднего в генеральной совокупности в единицах стандартной ошибки среднего:

$$z = \frac{\overline{X} - \mu}{\frac{\sigma}{\sqrt{(n)}}},$$

то эта величина z будет иметь стандартное нормальное распределение со средним равным нулю и стандартным отклонением равным единице.

Обратим внимание, что для расчета стандартной ошибки мы используем именно стандартное отклонение в генерально совокупности - σ. Ранее мы уже обсуждали, что на практике σ нам практически никогда не известна, и для расчетов стандартной ошибки мы используем выборочное стандартное отклонение.

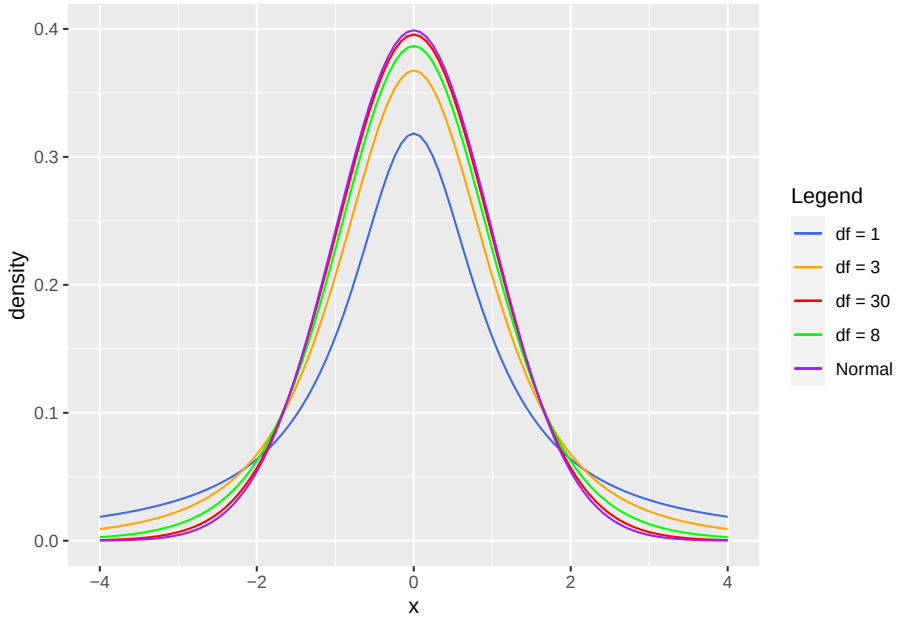
Строго говоря в таком случае распределение отклонения выборочного среднего и среднего в генеральной совокупности, деленного на стандартную ошибку, теперь будет описываться именно при помощи t-распределения.

$$t = \frac{\overline{X} - \mu}{\frac{sd}{\sqrt(n)}},$$

Таким образом, в случае неизвестной σ мы всегда будем иметь дело с t-распределением. На этом этапе возникает вопрос, почему в предыдущей главе использовался z-критерий для проверки гипотез, используя выборочное стандартное отклонение?

Мы уже знаем, что при довольно большом объеме выборки ($n > 30$) t-распределение совсем близко подбирается к нормальному распределению:

```
t_student_demo_plot
```



Поэтому иногда для простоты расчетов говорится, что если $n > 30$, то мы будем использовать свойства нормального распределения для наших целей. Строго говоря, это, конечно, неправильный подход, который часто критикуют. В до компьютерную эпоху этому было некоторое объяснение, чтобы не рассчитывать для каждого n большего 30 соответствующее критическое значение t-распределения, статистики как бы округляли результат и использовали нормальное распределение для этих целей. Сегодня с этим больше проблем нет и все статистические программы без труда рассчитывают все необходимые показатели для t-распределения

с любым числом степеней свободы. Действительно при выборках очень большого объема t-распределение практически не будет отличаться от нормального, однако хоть и очень малые различия все равно будут.

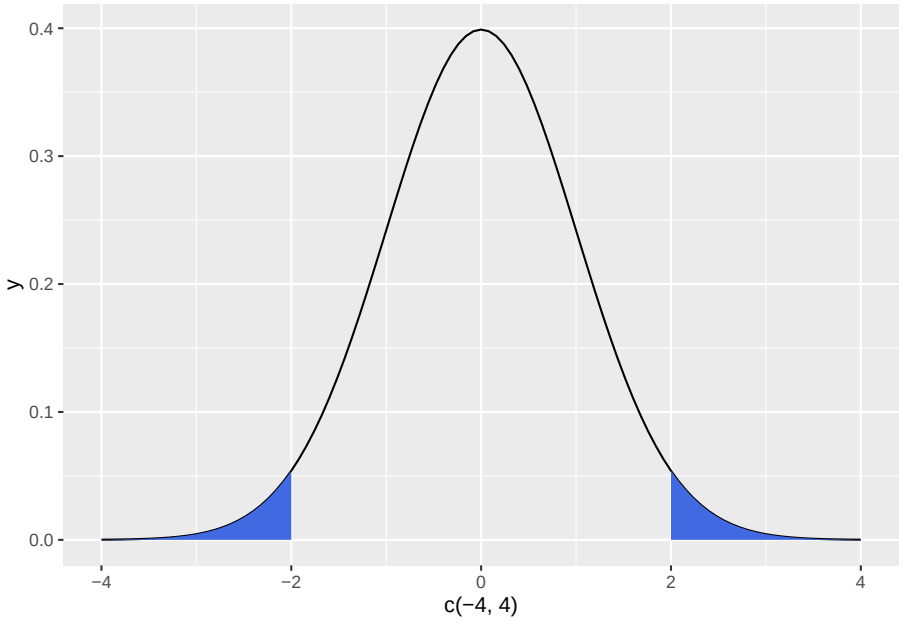
Поэтому, правильнее будет сказать, что мы используем t-распределение не потому что у нас маленькие выборки, а потому что мы не знаем стандартное отклонение в генеральной совокупности. Поэтому в дальнейшем мы всегда будем использовать t-распределение для проверки гипотез, если нам неизвестно стандартное отклонение в генеральной совокупности, необходимое для расчета стандартной ошибки, даже если объем выборки больше 30.

Если мы допустили, что все выборочные средние будут распределены нормальным образом, то вероятность получить отклонение превышающее 2σ как в левую, так и в правую сторону будет составлять:

```
pnorm(-2) + pnorm(2, lower.tail = FALSE)
```

```
## [1] 0.04550026
```

```
ggplot() +
    xlim(-4, 4) +
    stat_function(fun = dnorm) +
    geom_area(
        stat = "function", fun = dnorm,
        fill = "royalblue", xlim = c(-4, -2), aes(c(-4, 4))
    ) +
    geom_area(
        stat = "function", fun = dnorm,
        fill = "royalblue", xlim = c(2, 4), aes(c(-4, 4))
    )
```

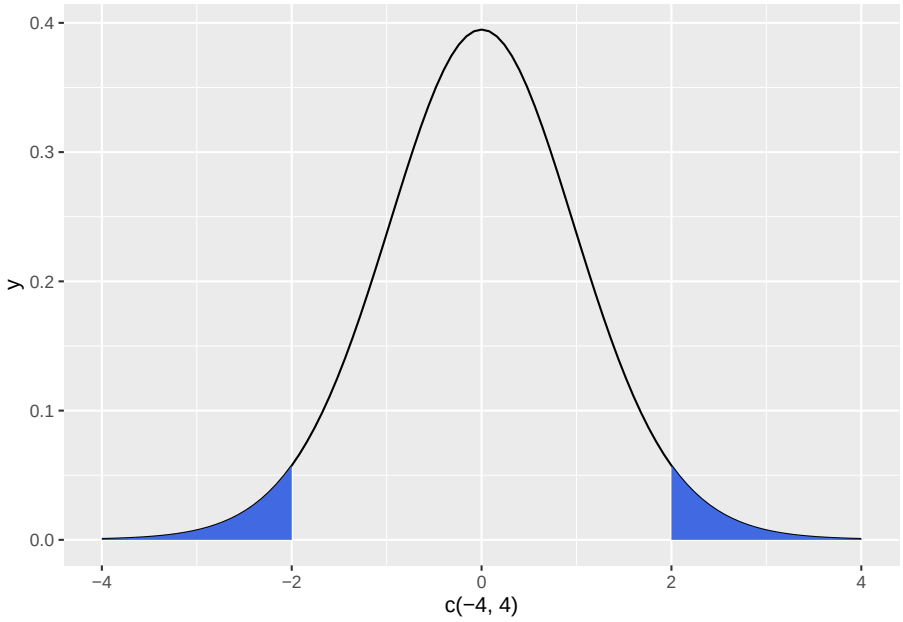


То есть p-уровень значимости будет меньше чем 0.05 и мы смело сможем отклонить нулевую гипотезу, согласно которой наша выборка принадлежит генеральной совокупности со среднем значением 10.8. Но как мы сказали при небольшом объеме выборки распределение выборочного среднего будет отличаться от нормального и вероятность получить более выраженное отклонение от среднего станет выше. Рассчитаем данную вероятность предположив, что мы работаем с t-распределением с 24 степенями свободы.

```
pt(-2, df = 24) + pt(2, df = 24, lower.tail = FALSE)
```

```
## [1] 0.05693985
```

```
ggplot() +
    xlim(-4, 4) +
    stat_function(fun = dt, args = list(df = 24)) +
    geom_area(
        stat = "function", fun = dt, args = list(df = 24),
        fill = "royalblue", xlim = c(-4, -2), aes(c(-4, 4))
    ) +
    geom_area(
        stat = "function", fun = dt, args = list(df = 24),
        fill = "royalblue", xlim = c(2, 4), aes(c(-4, 4))
    )
```



Это означает, что если бы мы пользовались t-распределением, то нулевую гипотезу мы бы отклонить не смогли. t-критерий рассчитывается так же как z-критерий:

$$t = \frac{\overline{x} - M}{\frac{sd}{\sqrt{(n)}}},$$

Однако если бы мы в этом случае получили 2, то в t-распределении с 24 степенями свободы $p = 0.056$ и нулевую гипотезу мы бы отклонить не смогли.

2.1.3 Понятие числа степеней свободы

Как уже понятно t-распределение зависит от числа наблюдений в выборке как $n-1$. Если дать более общее определение, то число степеней свободы - это число элементов которые могут варьироваться при расчете некоторого статистического показателя.

2.2 Сравнение двух средних, t-критерий Стьюдента

2.2.1 Сравнение двух средних

Критерий который позволяет сравнивать между собой два выборочных средних называется парным t-тестом или просто t-критерием Стьюдента.

2.2.2 t-критерий Стьюдента

Определение 2.2.1 *Критерий который позволяет сравнивать между собой два выборочных средних называется* ***парным t-тестом*** *или просто* ***t-критерием Стьюдента****.*

2.2.3 t-критерий Стьюдента

Предположим мы хотим сравнить два средних выборочных значения $\overline{x_1}$ рассчитанное на выборке со стандартным отклонением sd_1 и числом элементов выборки n_1 и $\overline{x_2}$ с sd_2 и n_2. Сначала сформулируем статистические гипотезы:

- H_0 - в генеральной совокупности никакого различия между этими значениями нет и $\mu_1 = \mu_2$.
- H_0 - $\mu_1 \neq \mu_2$.

Предположим, что верна нулевая гипотеза. Если это так, то при многократном повторении эксперимента и каждый раз рассчитывали разность между двумя выборочными средними значениями $\overline{x_1} - \overline{x_2}$, то эта величина распределилась бы следующим образом: мы бы получили симметричное распределение с средним значением $\mu_1 - \mu_2 = 0$ и стандартной ошибкой $se = \sqrt{\frac{sd_1^2}{n_1} + \frac{sd_2^2}{n_2}}$. Причем это распределение будет t-распределением с числом степеней свобод, вычисляемым по формуле $df = n_1 + n_2 - 2$. На основе этой информации мы можем рассчитать насколько далеко конкретно наша разность между двумя средними значениями отклонилась от предполагаемого показателя генеральной совокупности и тем самым рассчитать вероятность получить такие или еще более сильные отклонения при условии, что на самом деле нулевая гипотеза верна.

Окончательная формула для t-критерия будет иметь вид:

$$t = \frac{(\overline{x_1} - \overline{x_2}) - (\mu_1 - \mu_2)}{\sqrt{\frac{\mathrm{sd}_1^2}{n_1} + \frac{\mathrm{sd}_2^2}{n_2}}} \tag{2.1}$$

На основе этого показателя и зная число степеней свобод ($df = n_1 + n_2 - 2$) мы можем рассчитать p-уровень значимости, который покажет нам вероятность получить такое или еще более выраженное отклонение при условии, что нулевая гипотеза верна.

Пример: Процесс денатурации ДНК представляет разрушение водородных связей между двумя цепями этой молекулы и очень сильно зависит от температуры, с которой мы воздействуем на молекулу.

Table 2.1: Температуры плавления ДНК (датасет ds)

1	2
84.7	57.2
105.0	68.6
98.9	104.4
97.9	95.1
108.7	89.9
81.3	70.8
99.4	83.5
89.4	60.1
93.0	75.7
119.3	102.0
99.2	69.0
99.4	79.6
97.1	68.9
112.4	98.6
99.8	76.0
94.7	74.8
114.0	56.0
95.1	55.6
115.5	69.4
111.5	59.5

При сравнении двух видов между собой были получены следующие различия в средней температуре плавления ДНК:

```
ds_f <- ds |> pivot_longer(everything(),
    names_to = "dna", values_to = "val"
)
ds_stats <- ds_f |>
    group_by(dna) |>
    summarise(m = mean(val), sd = sd(val), n = length(val))
kable(ds_stats)
```

dna	m	sd	n
1	100.815	10.2465	20
2	75.735	15.4581	20

Формулируем гипотезы:

- H_0: $\mu_1 = \mu_2$
- H_1: $\mu_1 \neq \mu_2$

Считаем t-критерий:

```
t <- ds_stats |>
    mutate(dv = sd^2 / n) |>
    summarise(
        m.m = reduce(m, ~ .x - .y),
        m.dv = sqrt(sum(dv))
    ) |>
    mutate(t = m.m / m.dv) |>
    pull()
```

Число степеней свобод:

```
df_val <- sum(ds_stats$n) - 2
df_val
```

```
## [1] 38
```

Считаем интересующую нас вероятность:

```
p <- pt(-t, df = df_val) + pt(t, df = df_val, lower.tail = FALSE)
p
```

```
## [1] 4.894703e-07
```

Через встроенный функционал можно сильно проще:

```
t.test(val ~ dna, data = ds_f, var.equal = TRUE)$p.value
```

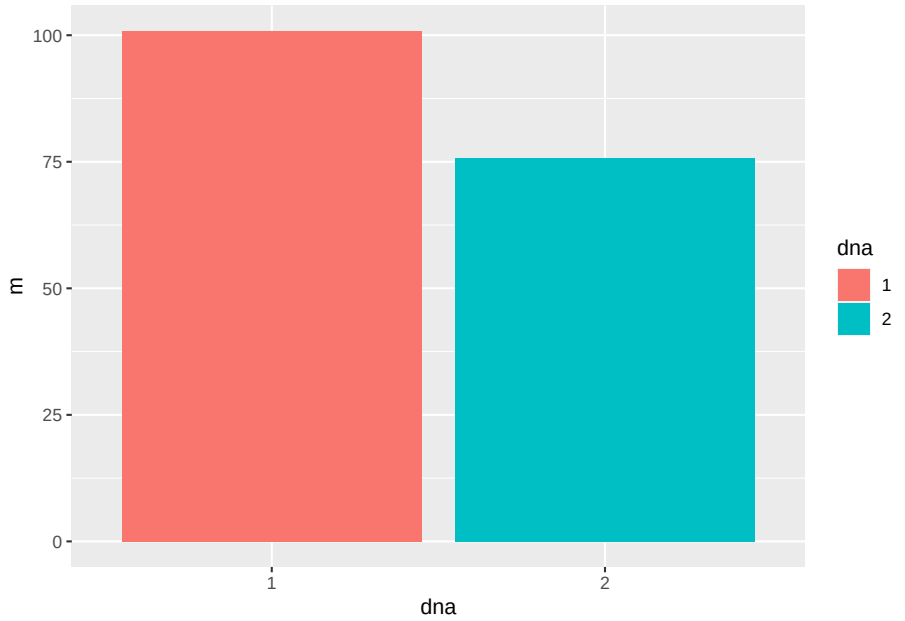
```
## [1] 4.894703e-07
```

Это меньше чем пороговое значение для p. Таким образом мы обнаружили статистически значимое различие в средней температуре плавления ДНК двух видов.

2.2.4 Построение графиков

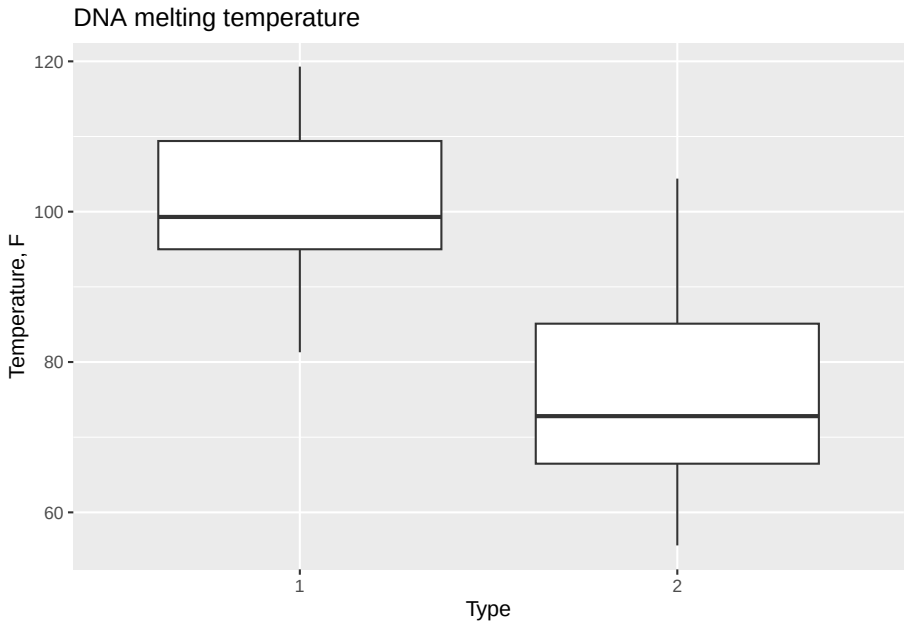
Для начала как делать не надо (что это вообще такое?):

```
ggplot(ds_stats, aes(x = dna, y = m, fill = dna)) +
    geom_col()
```



Как лучше:

```
ggplot(ds_f, aes(dna, val)) +
    geom_boxplot() +
    labs(
        title = "DNA melting temperature",
        x = "Type", y = "Temperature, F"
    )
```



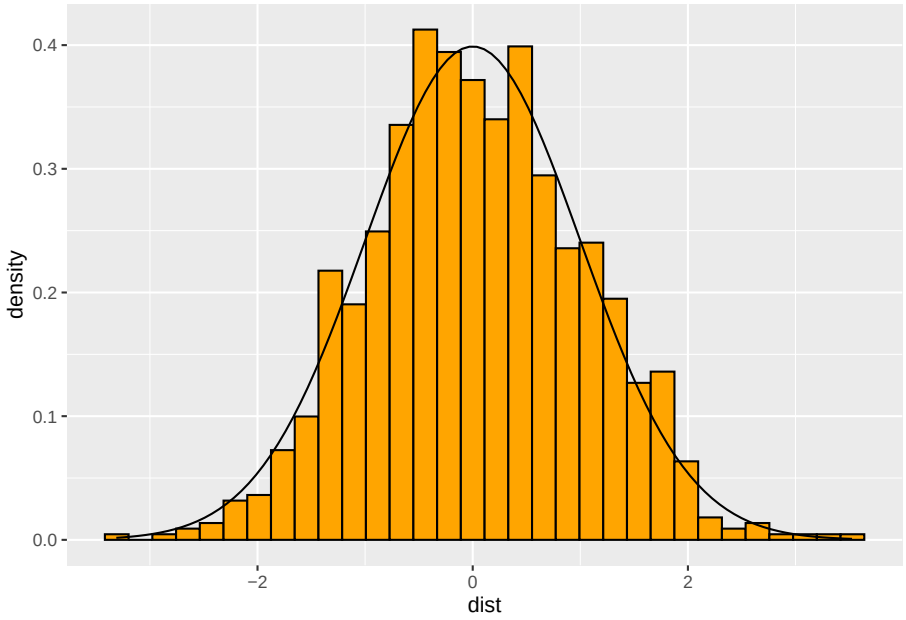
2.3 Проверка распределений на нормальность, QQ-Plot

2.3.1 Сравнение распределения с нормальным

Требование к нормальному распределению очень часто встречается в статистике при использовании различных методов. Как оценить отличие распределения от нормального?

Один из самых простых способов - построить гистограмму частот признака и поверх наложить кривую идеального нормального распределения. Например:

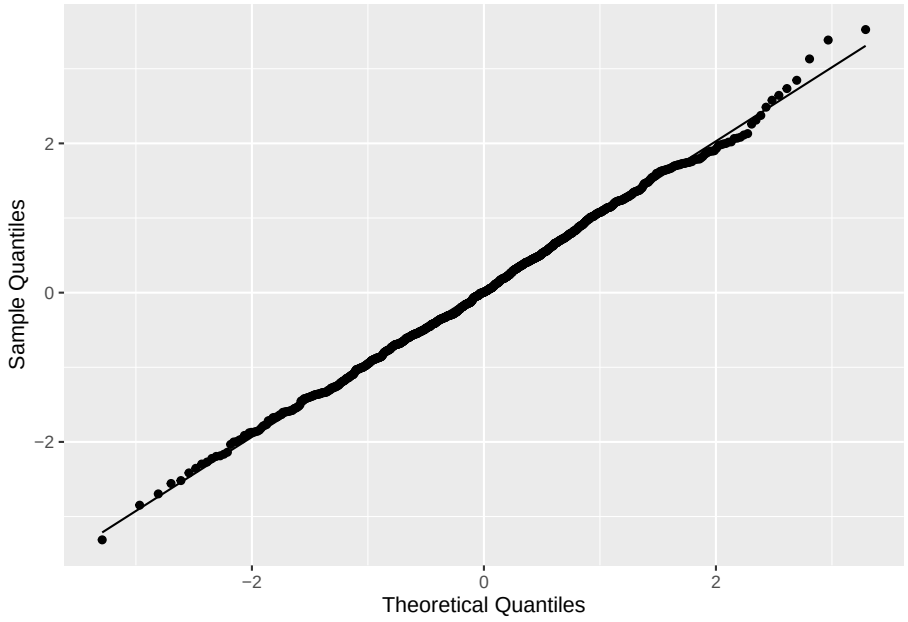
```
dist <- rnorm(1000)
ggplot(tibble(a = dist), aes(dist)) +
    geom_histogram(
        bins = 32, aes(y = after_stat(density)),
        fill = "orange", color = "black"
    ) +
    stat_function(fun = dnorm)
```



2.3.2 QQ-plot

Еще один графический способ проверить распределение на нормальность – QQ-plot.

```
ggplot(tibble(a = dist), aes(sample = dist)) +
    geom_qq() +
    geom_qq_line() +
    labs(x = "Theoretical Quantiles", y = "Sample Quantiles")
```



QQ-plot удобно использовать, когда число наблюдений невелико. Данных для построения гистограммы мало и удобно анализировать каждое значение отдельно что и возможно по такому графику.

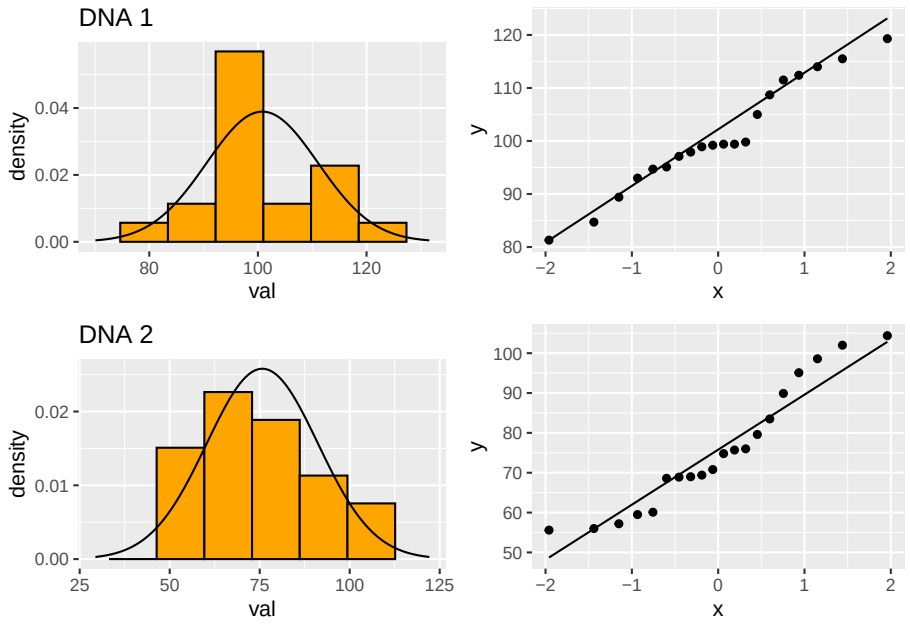
Попробуем проверить этот метод на задаче с температурой плавление ДНК у видов.

```
hist_verify_plot <- function(x) {
    m <- mean(x$val)
    std <- sd(x$val)
    dna_n <- unique(x$dna)
    return(x |> ggplot(aes(val)) +
        xlim((m - 3 * std), (m + 3 * std)) +
        geom_histogram(
            bins = 8, aes(y = after_stat(density)),
            color = "black", fill = "orange"
        ) +
        stat_function(fun = dnorm, args = list(
            mean = mean(x$val),
            sd = sd(x$val)
        )) +
        labs(title = str_c("DNA ", dna_n)))
}

verify_qq <- function(x) {
    return(
```

```
        x |> ggplot(aes(sample = val)) +
            geom_qq() +
            geom_qq_line()
    )
}

suppressWarnings(grid.arrange(
    hist_verify_plot(filter(ds_f, dna == 1)),
    verify_qq(filter(ds_f, dna == 1)),
    hist_verify_plot(filter(ds_f, dna == 2)),
    verify_qq(filter(ds_f, dna == 2)),
    nrow = 2, ncol = 2
))
```



2.3.3 Тест Шапиро-Уилка

Тест Шапиро-Уилка позволяет определить, что выборка изъята из генеральной совокупности и её распределение соответствует нормальному.

```
ds_f |>
    group_by(dna) |>
    summarise(
        swt = shapiro.test(val)$p.value
```

```
    ) |>
    kable()
```

dna	swt
1	0.6141727
2	0.1267972

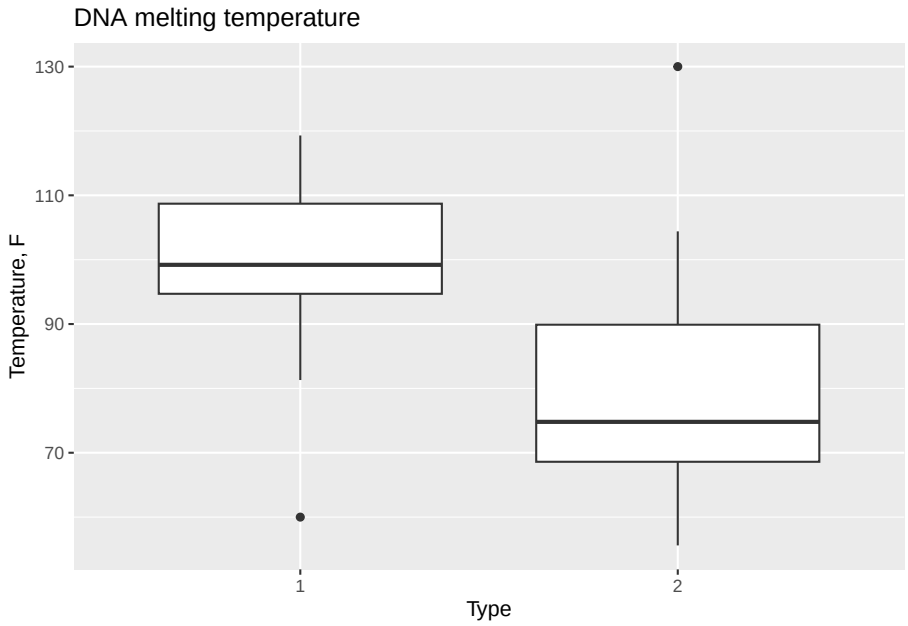
Если уровень SWT $>$ 0.05 это хорошо и не позволяет отклонить нулевую гипотезу.

2.3.4 Проблема выбросов

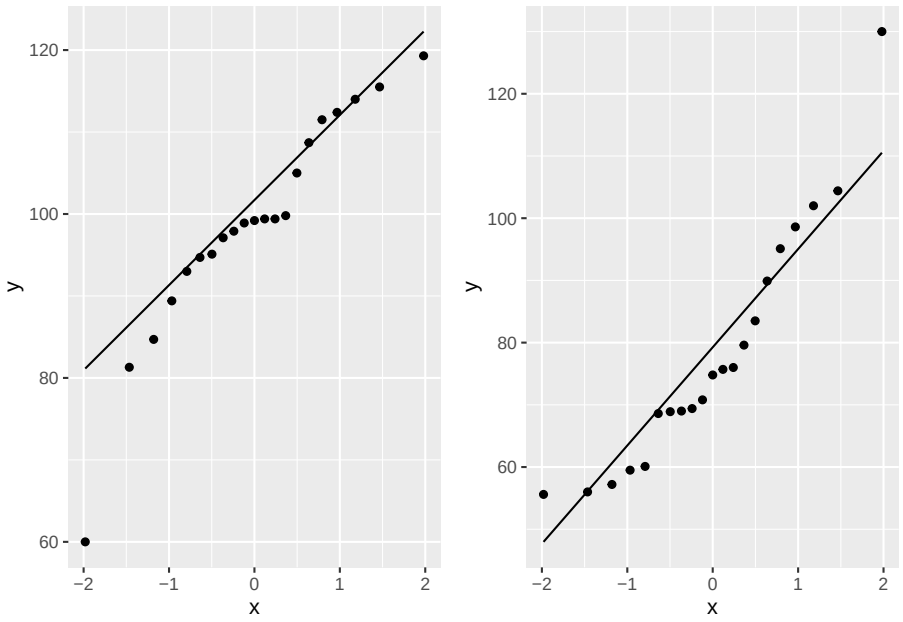
Посмотрим как влияет появление выбросов на параметры выборки. Для этого добавим по одному выбросу в каждый из наборов значений в задаче про температуру плавления и сравним результаты со старыми.

```
ds_damaged <- ds_f |> bind_rows(
    tibble(dna = c("1", "2"), val = c(60, 130))
)

ggplot(ds_damaged, aes(dna, val)) +
    geom_boxplot() +
    labs(
        title = "DNA melting temperature",
        x = "Type", y = "Temperature, F"
    )
```




```
suppressWarnings(grid.arrange(
    verify_qq(filter(ds_damaged, dna == 1)),
    verify_qq(filter(ds_damaged, dna == 2)),
    nrow = 1, ncol = 2
))
```




```
ds_damaged |>
    group_by(dna) |>
    summarise(
        m = mean(val),
        sd = sd(val),
        n = length(val),
        swt = shapiro.test(val)$p.value
    ) |>
    kable()
```

dna	m	sd	n	swt
1	98.87143	13.38163	21	0.0916505
2	78.31905	19.16321	21	0.0549744

2.3.5 U-критерий Манна-Уитни

Что делать если распределение признака сильно отличается от нормального и мы опасаемся применять t-признак Стьюдента? В таких ситуациях используется непараметрический аналог, называемый U-критерием Манна-Уитни.

```
ds_damaged |>
    mutate(
        dna = str_c("DNA", dna)
    ) |>
    group_by(dna) |>
    mutate(row = row_number()) |>
    pivot_wider(
        names_from = dna,
        values_from = val
    ) |>
    summarise(
        MW = wilcox.test(DNA1, DNA2, exact = FALSE)$p.value
    ) |>
    pull()
```

```
## [1] 0.0004489329
```

2.4 Однофакторный дисперсионный анализ

2.4.1 Расчет на практическом примере

Предположим, что у нас есть следующий набор данных:

```
ds <- tibble(
    "1" = c(3, 1, 2),
    "2" = c(5, 3, 4),
    "3" = c(7, 6, 5)
)
ds_f <- ds |>
    pivot_longer(
        cols = everything(),
        names_to = "name",
        values_to = "vals"
    )
kable(ds)
```

1	2	3
3	5	7
1	3	6
2	4	5

Сформулируем гипотезы:

- $H_0 : \mu_1 = \mu_2 = \mu_3$
- $H_1 : \mu_1 \neq \mu_2 \neq \mu_3$

Среднее по всему набору:

```
ds_f |>
    summarise(
        mean = mean(vals)
    ) |>
    pull()
```

```
## [1] 4
```

Введем понятие SST – общая сумма квадратов. Этот показатель позволяет увидеть насколько высока изменчивость данных.

```
ds_f |>
    summarise(
        sst = sum((vals - mean(vals))^2)
    ) |>
    pull()
```

```
## [1] 30
```

Число степеней свободы для всего набора данных будет составлять $8(n - 1)$. Так же есть два важных показателя: SSW и SSB. SSW – это сумма квадратов внутри выборки:

```
ssw <- ds_f |>
    group_by(name) |>
    summarise(
        ssw = sum((vals - mean(vals))^2)
    ) |>
    ungroup() |>
    summarise(
        ssw = sum(ssw)
    ) |>
    pull()
```

Число степеней свободы для SSW - это сумма степеней свободы для каждой группы. В данном случае соответственно - 6. Еще один показатель: SSB - сумма квадратов меж выборок. Вычисляется как:

```
ssb <- ds_f |>
    group_by(name) |>
    summarise(
        n = length(vals),
        mean = mean(vals)
    ) |>
    mutate(
        means_all = mean(ds_f$vals),
        pre_ssb = n * (mean - means_all)^2
    ) |>
    summarise(
        ssb = sum(pre_ssb)
    ) |>
    pull()
```

2.4.2 F-значение

Введем так же F-value – показатель Фишера. Он вычисляется по формуле:

$$F = \frac{SSB/df_{SSB}}{SSW/df_{SSW}} \tag{2.2}$$

В нашем случае будет иметь значение:

```
f <- (ssb / 2) / (ssw / 6)
f
```

```
## [1] 12
```

Отметим так же, что отношение $\frac{SSB}{ds_{SSB}} \to 0$ с ростом числа элементов, из чего следует, что $F \to 0$ с ростом числа элементов. Наконец, основываясь на значении показателя Фишера, мы так же можем оценить отклонение значений от нулевой гипотезы:

```
pf(f, df1 = 2, df2 = 6, lower.tail = FALSE)
```

```
## [1] 0.008
```

Или через готовую функцию в языке R:

```
one_way <- aov(vals ~ name, data = ds_f) |> tidy()
one_way |> kable()
```

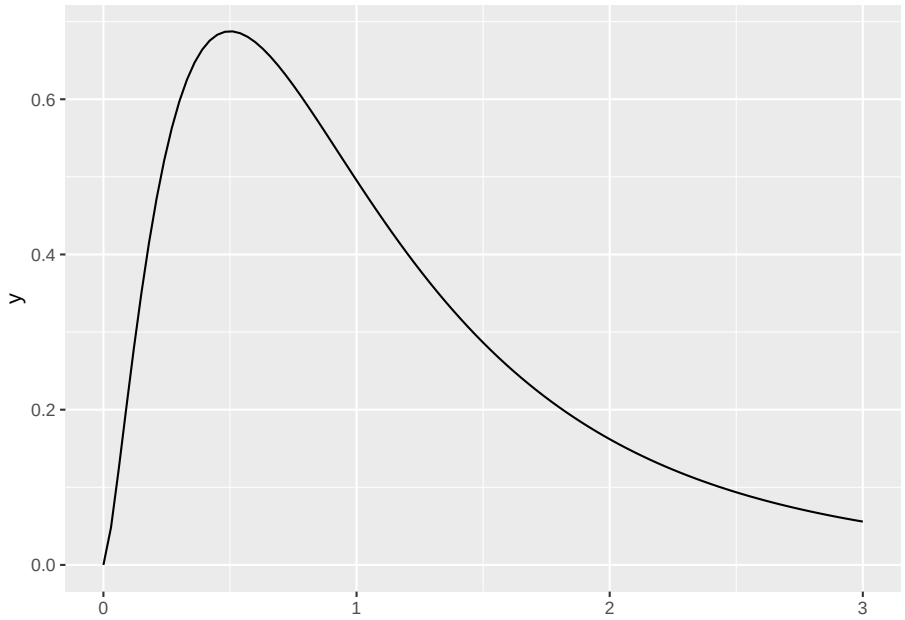
term	df	sumsq	meansq	statistic	p.value
name	2	24	12	12	0.008
Residuals	6	6	1	NA	NA

```
drop_na(one_way)$p.value
```

```
## [1] 0.008
```

В целом ANOVA позволяет сравнивать значимости различий для произвольного количества групп. Характерный вид распределения Фишера представлен ниже. Так же отметим, что он считается в одном направлении, поскольку само распределение Фишера имеет только положительные значения.

```
ggplot() +
    xlim(0, 3) +
    stat_function(
        fun = stats::df,
        args = list(df1 = 5, df2 = 10)
    )
```



2.4.3 Применение и интерпретация

Генотерапия позволяет корректировать работу дефектного гена, ответственного за развитие заболевания. В эксперименте сравнивалась эффективность четырех различных типов терапии. Результаты исследования представлены в таблице:

```
genetherapy <- read_csv("genetherapy.csv", show_col_types = FALSE)
gene_descr <- genetherapy |>
    group_by(Therapy) |>
    summarise(
        N = length(expr),
        M = mean(expr),
        SD = sd(expr)
    )
kable(gene_descr)
```

Therapy	N	M	SD
A	15	99.73333	4.165619
B	15	98.80000	5.894307
C	15	94.40000	5.193402
D	15	92.33333	3.735289

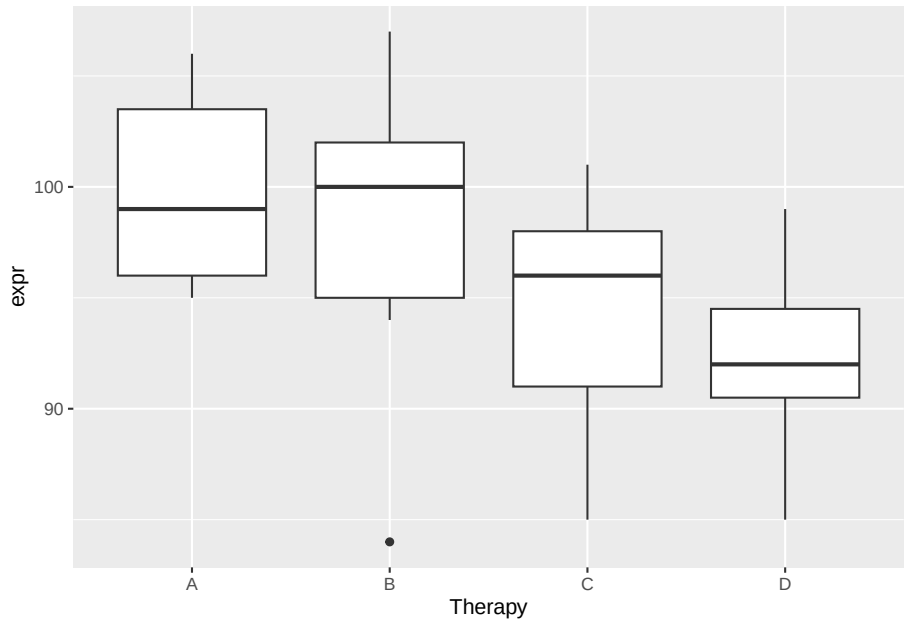
Вводим гипотезу: $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4$. Применим ANOVA:

```
gene_anova <- aov(expr ~ Therapy, data = genetherapy) |> tidy()
kable(gene_anova)
```

term	df	sumsq	meansq	statistic	p.value
Therapy	3	560.7167	186.90556	8.037302	0.0001525
Residuals	56	1302.2667	23.25476	NA	NA

Здесь sumsq то же самое, что и SSB. meansq – $\frac{SSB}{df}$. Как видим по результатам p.value – нулевая гипотеза отклоняется (хотя бы 2 средних отличаются между собой). Теперь посмотрим график:

```
genetherapy |>
    ggplot(aes(Therapy, expr)) +
    geom_boxplot()
```



2.5 Множественные сравнения в ANOVA

2.5.1 Проблема множественного сравнения выборок

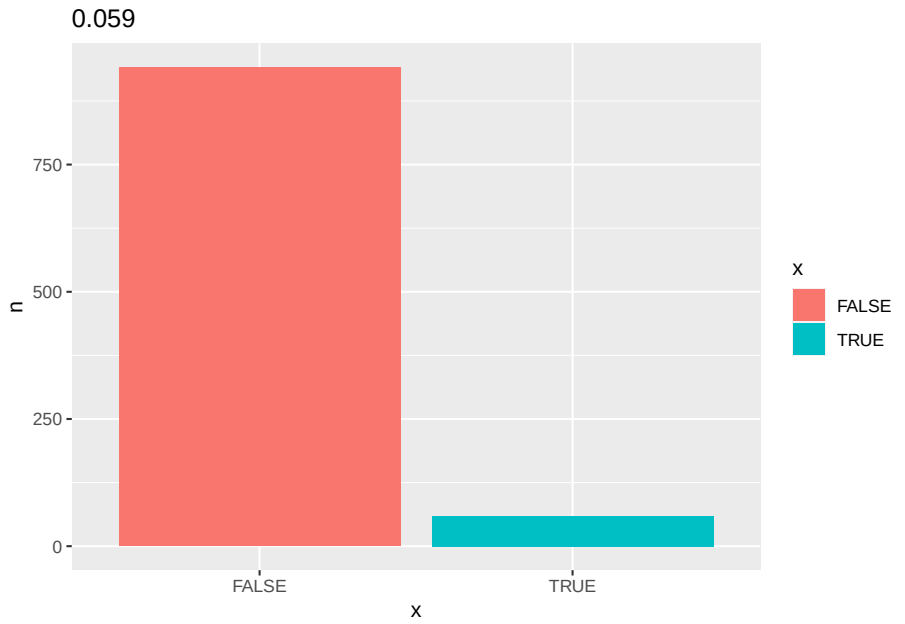
Проблема множественного сравнения возникает когда нужно сравнить множество выборок между собой. Почему при этом нельзя попарно использовать критерий Стьюдента? Для этого приводим пример: Пусть есть генеральная совокупность со средним 0 и стандартным отклонением 1. Из этой совокупности мы будем многократно извлекать выборки и сравнивать их между собой. Функция для изъятия выборок имеет следующий вид:

```
false_alarm <- function(m, n, a) {
    tt_test <- function(x, y) {
        return(t.test(x, y, var.equal = TRUE)$p.value)
    }
    d <- matrix(0, n, m)
    s <- combn(m, 2)
    x <- vector("logical", 1000)
    for (i in 1:1000) {
        d <- apply(d, 2, function(x) rnorm(n))
        for (j in 1:ncol(s)) {
            test <- tt_test(d[, s[1, j]], d[, s[2, j]])
```

```
            if (test <= a) {
                x[i] <- TRUE
                break
            }
        }
    }
    tx <- tibble(x = x) |>
        group_by(x) |>
        count()
    tx |> ggplot(aes(x = x, y = n, fill = x)) +
        geom_col() +
        labs(
            title = tx |>
                filter(x == TRUE) |>
                select(c(n)) |>
                pull() / 1000
        )
}

false_alarm(2, 30, 0.05)
```

```
## Adding missing grouping variables: `x`
```

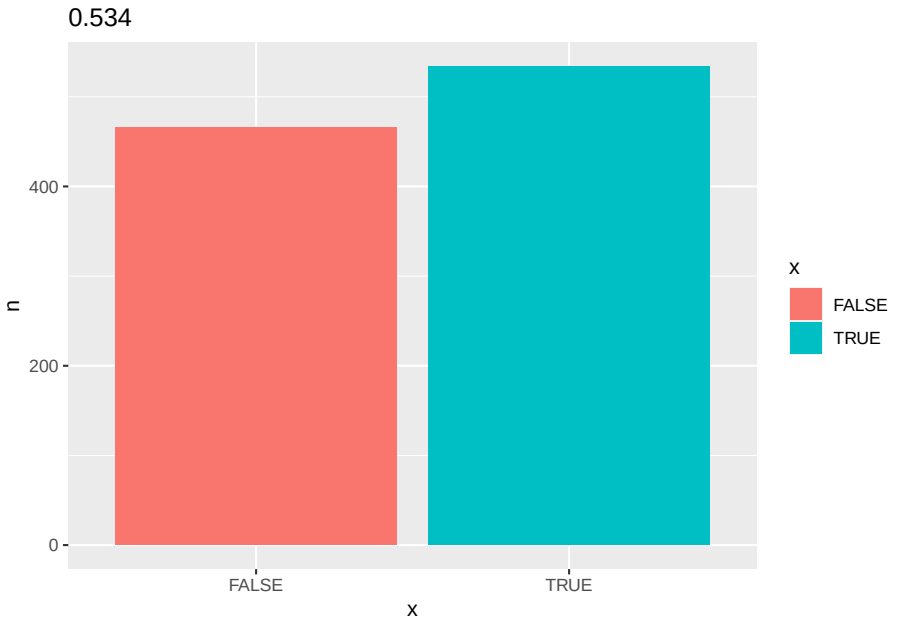


На графике показано в каком проценте случаев мы получили статистически

значимые различия. В 5% случаев мы получили статистически значимые различия из одной выборки. То есть на самом деле никаких статистически значимых различий мы не должны были получить. Но поскольку мы выбрали некоторый порог p-уровня значимости после которого мы принимаем различия достоверными. Теперь увеличим количество выборок:

```
false_alarm(8, 30, 0.05)
```

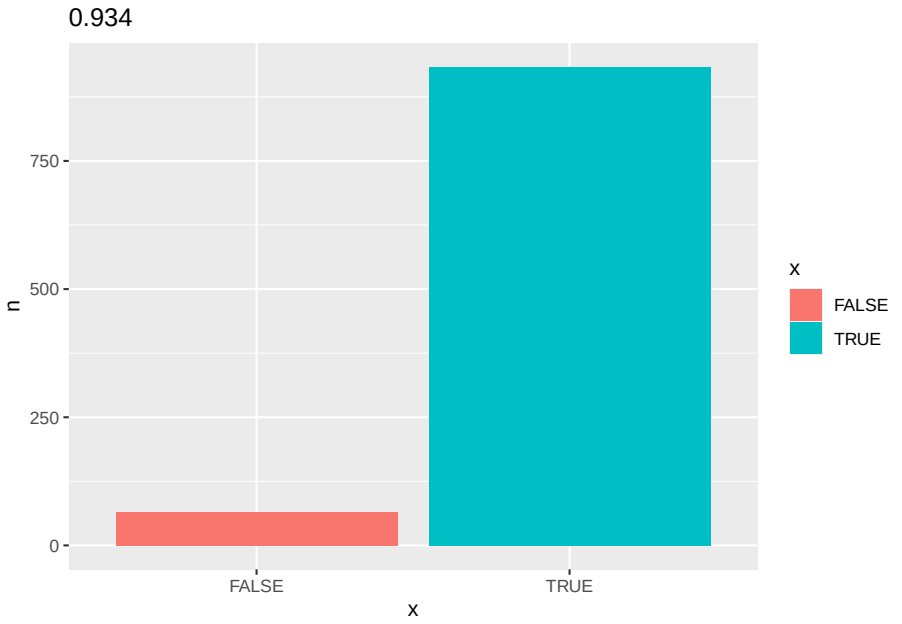
```
## Adding missing grouping variables: `x`
```



Теперь уже в 52% случаев мы получим хотя бы одно статистически значимое различие между 8 выборками несмотря на то, что они из одной генеральной совокупности. Теперь посмотрим что будет если бы сделаем сравнение в 20 выборках:

```
false_alarm(20, 30, 0.05)
```

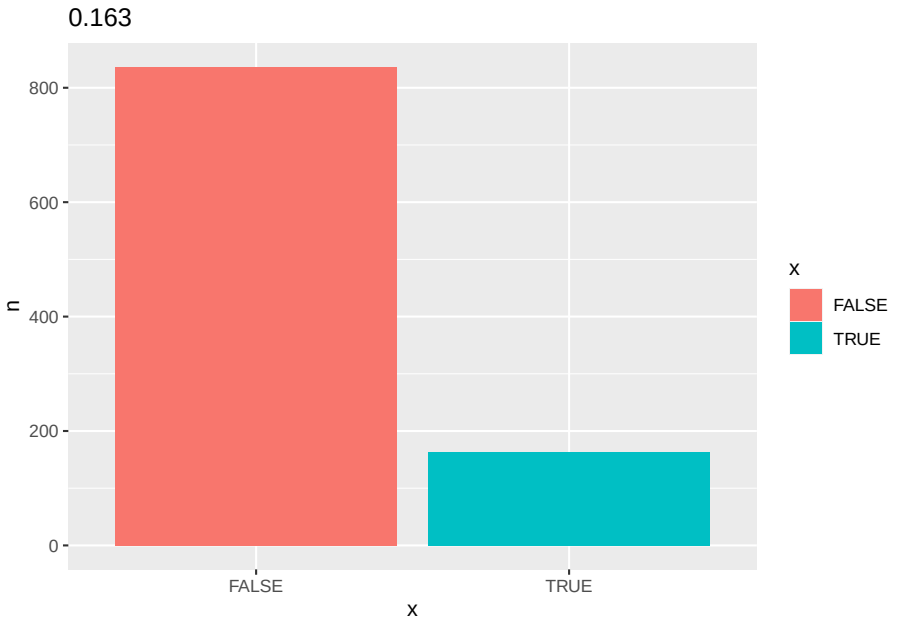
```
## Adding missing grouping variables: `x`
```



Когда мы сравниваем между собой 2 группы мы принимаем различия значимыми если показатель p меньше 0.05. То есть даже если различий на самом деле нет, в 5% случаев мы бы получили различия случайно. Но когда мы сравниваем 4 комбинации оказывается, что вероятность получить одно различие уже значительно больше. Если мы увеличим количество групп, то вероятность получить различия будет стремиться к единице. Это означает, что если мы многократно увеличиваем количество сравниваемых групп, то вероятность получить хотя бы одно различие, которого на самом деле вообще не существует очень сильнол увеличивается. Таким образом нам нужно корректировать выбор p-уровня. Такой поправкой является поправка Бонферрони.

```
false_alarm(20, 30, 0.05 / 28)
```

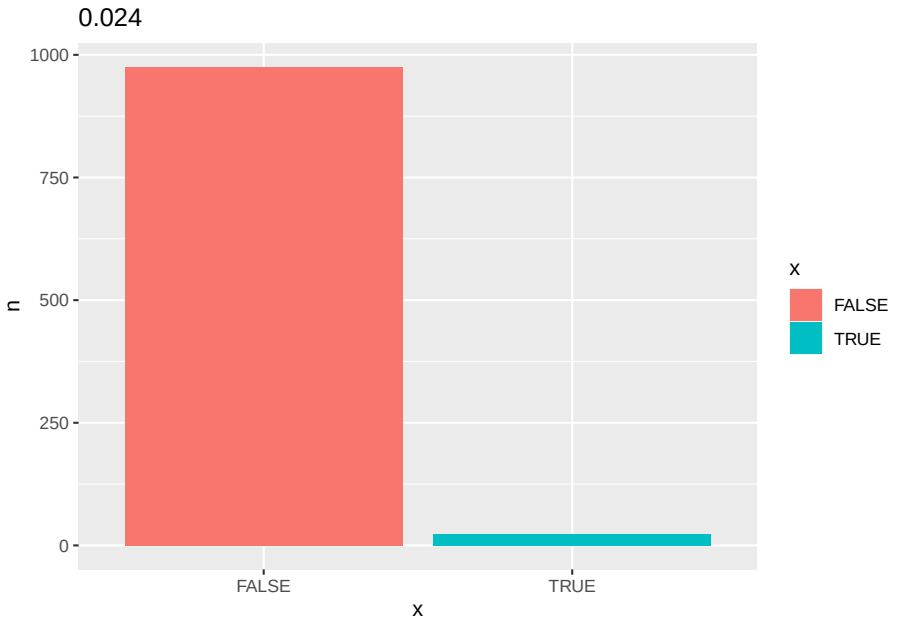
```
## Adding missing grouping variables: `x`
```



Мы вернулись к тому же уровню, что и при 2 сравнениях. Аналогичная ситуация будет и при 20 группах:

```
comb_count <- ncol(combn(20, 2))
false_alarm(20, 30, 0.05 / comb_count)
```

```
## Adding missing grouping variables: `x`
```



При достаточном уровне терпения можно всегда получить значимые различия. Допустим мы не смогли отвергнуть нулевую гипотезу и стали добавлять большое число других факторов (пол, место рождения, возраст, семейное положение и тд) и продолжали сравнивать испытуемых по уровню экспрессии гена и на определенном этапе получим значимую связь у испытуемых по месту рождения и сделать выводы о влиянии внешней среды. Проблема в том, что при поправке Бонферрони при 100 сравнениях гарантируется отсутствие хотя бы одного ложного результата, но при этом упускается около 80% реальных открытий. Поэтому она сильно критикуется: она так понижает уровень значимости, что получить различия зачастую становится невозможным. Но современная статистика вынуждена работать с большим количеством сравнений. Поэтому разработана серия методов, которые работают лучше и возволяет не так сильно понижать p-уровень.

2.5.2 Критерий Тьюки

Возвращаемся к примеру со сравнением 4х типов терапии. Критерий Тьюки очень похож на TTest, однако инчае рассчитывается стандартная ошибка. С помощью некго можно рассчитать доверительный интервал между средними значениями

$$\overline{x_A} - \overline{x_B}$$

и если такой доверительный интервал не включает в себя 0, то можно отклонить нулевую гипотезу о равенстве средних.

```
gene_anova_tukey <- aov(expr ~ Therapy, data = genetherapy) |>
    TukeyHSD() |>
    tidy() |>
    select(-c(null.value))
kable(gene_anova_tukey, digits = 4)
```

term	contrast	estimate	conf.low	conf.high	adj.p.value
Therapy	B-A	-0.9333	-5.5959	3.7292	0.9514
Therapy	C-A	-5.3333	-9.9959	-0.6708	0.0189
Therapy	D-A	-7.4000	-12.0626	-2.7374	0.0005
Therapy	C-B	-4.4000	-9.0626	0.2626	0.0710
Therapy	D-B	-6.4667	-11.1292	-1.8041	0.0029
Therapy	D-C	-2.0667	-6.7292	2.5959	0.6458

Видим, что значимо отличаются группы C-A, D-A, D-B:

```
gene_anova_tukey |>
    filter(adj.p.value < 0.05) |>
    kable(digits = 4)
```

term	contrast	estimate	conf.low	conf.high	adj.p.value
Therapy	C-A	-5.3333	-9.9959	-0.6708	0.0189
Therapy	D-A	-7.4000	-12.0626	-2.7374	0.0005
Therapy	D-B	-6.4667	-11.1292	-1.8041	0.0029

2.5.3 Интерпретация результатов

Таким образом, если мы сравнили множество групп и не применили поправку, то получаем критику в свой адрес. Если мы применим поправку Бонферрони и остались те же значимые различия, то никаких претензий быть не может. Так же есть более свободные поправки. Более важный вопрос - это зачем вообще производится сравнение. Если мы проверяем какую-то гипотезу и не нашли подтверждений, то можно додавить расчеты так, чтобы что-то там все таки отыскать. Надо так же смотреть на дополнительные факторы. Зачастую значимые различия возникают за счет множественного сравнения, так что нужно применять поправку.

2.6 Многофакторный ANOVA

2.6.1 Двухфакторный дисперсионный анализ

Задача: Атеросклероз довольно опасное заболевание - причина ишемичной болезни сердца и инсультов. Анализ экспрессии генов лейкоцитов позволяет предсказать вероятность развития данного заболевания. В эксперименте исследовался уровень экспрессии в зависимости от возраста пациентов и дозировки лекарства аторвастатина.

```
data_ath <- read_csv("atherosclerosis.csv",
    show_col_types = FALSE
) |>
    mutate(
        age = factor(case_when(
            age == 1 ~ "Young",
            age == 2 ~ "Old",
        )),
        dose = factor(case_when(
            dose == "D1" ~ "High",
            dose == "D2" ~ "Low",
        ))
    )
ath_stat <- data_ath |>
    group_by(age, dose) |>
    summarise(
        N = dplyr::n(),
        Mx = mean(expr),
        SD = sd(expr)
    )
```

```
## `summarise()` has grouped output by 'age'. You can override
## using the `.groups` argument.
```

```
kable(ath_stat)
```

age	dose	N	Mx	SD
Old	High	16	101.0048	5.116310
Old	Low	16	102.2736	5.135375
Young	High	16	104.7585	5.863454
Young	Low	16	105.5459	4.369024

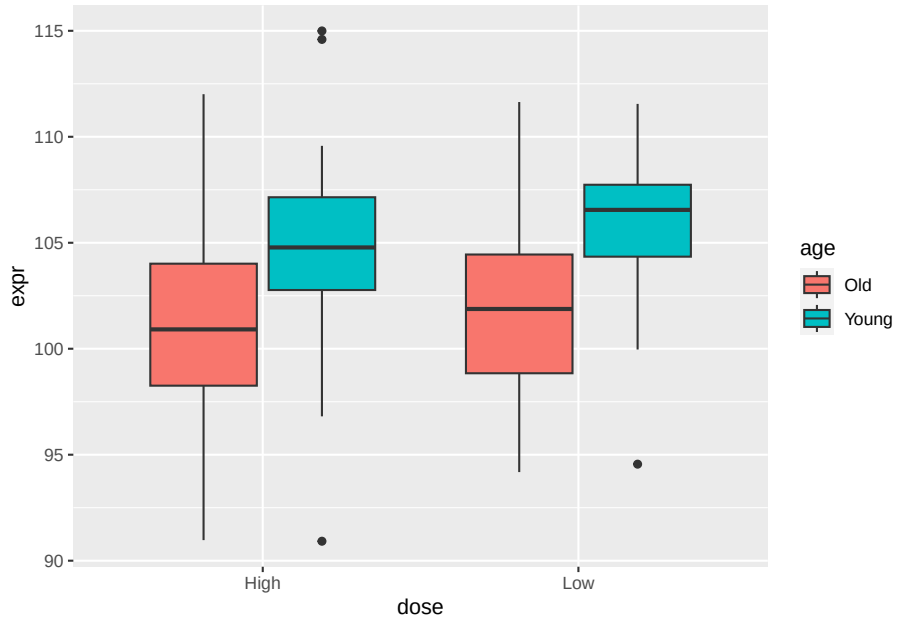
Результаты дисперсионного анализа:

```
ath_anova <- aov(expr ~ age + dose, data = data_ath) |> tidy()
kable(ath_anova)
```

term	df	sumsq	meansq	statistic	p.value
age	1	197.45275	197.45275	7.5695923	0.0078044
dose	1	16.91224	16.91224	0.6483514	0.4238302
Residuals	61	1591.18450	26.08499	NA	NA

Теперь строим график и интерпретируем результат:

```
data_ath |>
    ggplot(aes(dose, expr, fill = age)) +
    geom_boxplot()
```



Значимым является только фактор возраста (обращаем внимание, что в ANOVA p-уровень ниже 0.05 только у группы по возрасту) потому что вне зависмости от дозировки, молодые участники оказались с более высоким фактором переменной чем пожилые. Таким образом препарат значимый для фактора возраста, но не значимый для фактора дозировки. Возможна ситуация, когда значимы оба фактора.

2.6.2 Взаимодействие факторов в ANOVA

Чтобы познакомиться с взаимодействием факторов рассмотрим еще один пример: Исследователей интересовало влияние инъекции некоторого гормона на показатель концентрации кальция в плазме крови у птиц с учетом их пола. В таблице представлены данные экспериментальной и контрольной группы.

```
data_birds <- read_csv("birds.csv",
    show_col_types = FALSE
) |>
    mutate(
        hormone = factor(case_when(
            hormone == 0 ~ "Hormone 1",
            hormone == 1 ~ "Hormone 2",
        )),
        sex = factor(case_when(
            sex == 0 ~ "Male",
            sex == 1 ~ "Female"
        ))
    )
birds_stat <- data_birds |>
    group_by(hormone, sex) |>
    summarise(
        N = dplyr::n(),
        Mx = mean(var4),
        SD = sd(var4)
    )
```

```
## `summarise()` has grouped output by 'hormone'. You can
## override using the `.groups` argument.
```

```
birds_stat |> kable()
```

hormone	sex	N	Mx	SD
Hormone 1	Female	16	17.60892	2.449753
Hormone 1	Male	16	19.88729	3.677211
Hormone 2	Female	16	19.74366	3.383574
Hormone 2	Male	16	17.29225	2.864428

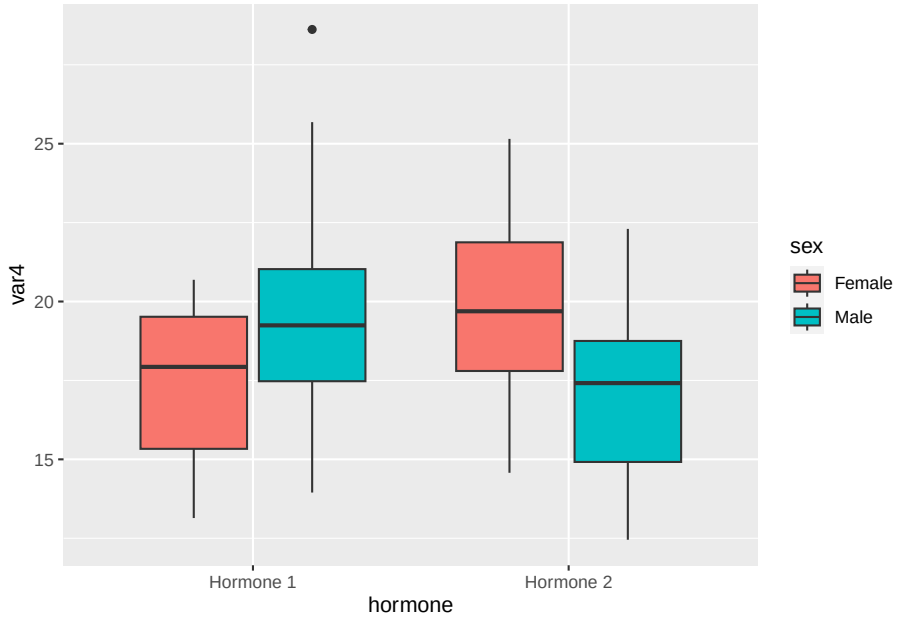
Результаты дисперсионного анализа:

```
birds_anova <- aov(
    var4 ~ hormone + sex + hormone * sex,
    data = data_birds
) |> tidy()
kable(birds_anova)
```

term	df	sumsq	meansq	statistic	p.value
hormone	1	0.8474716	0.8474716	0.0865281	0.7696534
sex	1	0.1197625	0.1197625	0.0122279	0.9123185
hormone:sex	1	89.4833843	89.4833843	9.1363898	0.0036818
Residuals	60	587.6503942	9.7941732	NA	NA

Мы видим, что ни фактор инъекции, ни фактор пола не оказали значимого влияния на зависимую переменную. Изменчивость их невелика. Взаимодействие оказало значительное влияние. Если мы построим график наших результатов, то увидим следующую картину:

```
data_birds |>
    ggplot(aes(hormone, var4, fill = sex)) +
    geom_boxplot()
```



Сам факт инъекции по разному повлиял на концентрацию кальция в плазме в зависимости от пола. В случае мужского пола это привело к

увеличению интересующего нас показателя и к снижению концентрации в случае женского пола. В этом и заключается идея взаимодействия факторов - когда некоторые переменные оказывают различное влияние на интересующий нас показатель в зависимости от уровней или градации другой независимой переменной.

2.6.3 Требования к данным

Также важно отметить некоторые требования к данным. Дисперсионный анализ требует нормальность распределения в каждой из групп и гомогенность дисперсий - то есть требование, чтобы дисперсия была примерно одинкаовой в каждой из групп. Приятный факт, что ANOVA достаточно устойчива к обоим этим нарушениям. Но если наблюдений меньше 30 в каждой из групп лучше проверять данные на нормальность распределения и на гомогенность дисперсии. Для проверки гомогенности можно построить BoxPlot и убедиться в отсутствии большого числа выбросов или применить тест Левена и если дисперсии равны, то все хорошо.

2.6.4 Интерпретация результатов

Само по себе применение дисперсионного анализа не дает оснований говорить о причинной зависимости данных. Например, если мы решим выяснить кто лучше разбирается в статистике - математики, филологи или психологи, и для этого применим дисперсионный анализ, то значимые различия между группами (например математики будут разбираться лучше всего) может означать как тот факт, что те кто занимается математикой лучше научились статистике и теперь лучше её понимают, так и тот, что те, кто лучше понимает статистику становятся математиками, а не филологами или психологами. Дисперсионный анализ проверяет гипотезу о взаимосвязи номинативной профессии и зависимой переменной (средней успеваемости по статистике). Поэтому всегда важно задаваться вопросом “а можно ли объяснить данные с другой стороны”.

2.7 АБ тесты и статистика

Мы подробно изучили теоретические аспекты статистических методов. Пришло время узнать, как статистика применяется на практике для реальных исследований и экспериментов. АБ тестирование - это проведение экспериментов при помощи статистики, пожалуй, самый яркий пример того, зачем статистика нужна в реальной жизни! A/B тесты - один из основных инструментов в продуктовой аналитике. Этот метод

маркетингового исследования заключается в том, что контрольная группа элементов сравнивается с набором тестовых групп, где один или несколько показателей изменены для того, чтобы выяснить, какие из изменений улучшают целевой показатель. Например, мы можем поменять цвет кнопки для регистрации с красного на синий и сравнить, насколько это будет эффективно. Данный раздел предлагается к просмотру на YouTube: https://www.youtube.com/watch?v=gljfGAkgX_o

Chapter 3

Корреляция и регрессия

3.1 Понятие корреляции

Начнем знакомствос таким понятием, как корреляция. При помощи корреляции мы научимся исследовать взаимосвязь двух количественных переменных, узнаем что такое положительная и отрицательная корреляции, разберемся как найти коэффициент корреляции и о чем он нам говорит.

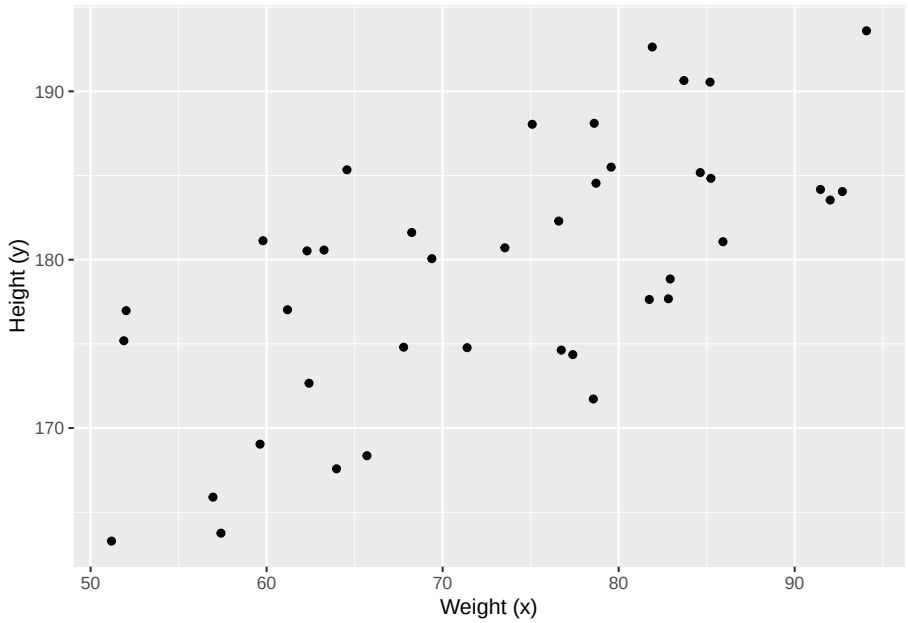
Рассмотрим несколько примеров.

Предположим мы захотели узнать как взаимосвязан возраст и физическая активность. Набрали выборку испытуемых и у каждого испытуемого узнали его возраст и число тренировок в неделю. Если нанести все точки на график, то легко заметить, что все значения числа тренировок с ростом возраста будут понижаться.

Такая взаиомвязь называется отрицательной корреляцией. Если с ростом одной переменной растет и вторая, то такая корреляция называется положительной. По диаграмме рассеивания мы можем понять насколько сильно взаимосвязаны наши переменные, однако нужно что-то более конкретное, показатель. Он называется коэффициентом корреляции. Найдем формулу для расчета. Для этого используем пример увеличения роста с увеличением веса. Рост обозначим за y, вес за x. График выглядит следующим образом:

```
wh_data <- tibble(
    weight = runif(40, 50, 95),
    height = runif(40, -10, 10) + 150
) |>
    mutate(
        height = 0.4 * weight + height
```

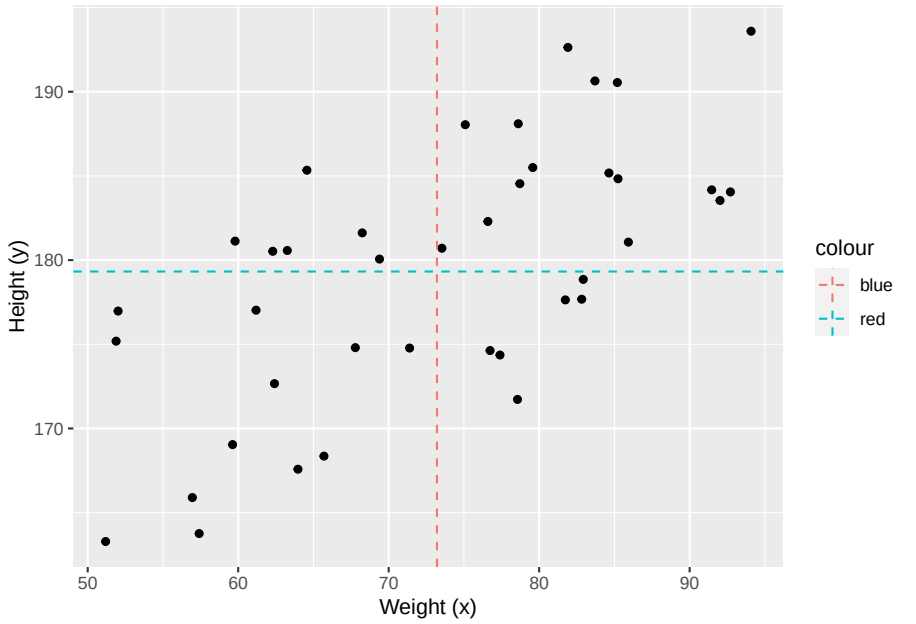
```
    )
wh_data |> ggplot(aes(x = weight, y = height)) +
    geom_point() +
    labs(
        x = "Weight (x)",
        y = "Height (y)"
    )
```



Добавим на график две линии: красная вертикальная будет соответствовать среднем значению веса, а синяя горизонтальная - среднему значению роста.

```
wh_data |> ggplot(aes(x = weight, y = height)) +
    geom_point() +
    labs(
        x = "Weight (x)",
        y = "Height (y)"
    ) +
    geom_hline(
        aes(
            yintercept = mean(height),
            color = "red"
        ),
        linetype = 2
    ) +
    geom_vline(
```

```
        aes(
            xintercept = mean(weight),
            color = "blue"
        ),
        linetype = 2
    )
```



Видим, что вся диаграмма разбилась линиями на 4 сектора. Если большая часть наших наблюдений находится в верхнем правом и в нижнем левом секторах, значит наща корреляция положительная, поскольку если человек находится в верхнем правом секторе, то он и тяжелее и выше среднего. Аналогичная ситуация со втором секторе. Теперь для каждой точки рассчитаем следующий параметр:

$$(x_i - \overline{X}) \cdot (y_i - \overline{Y})$$

Так как отклонение для точек в верхнем правом секторе положительное, то и показатель будет положительный. Для точек в нижнем левом отклонения отрицательные, значит и показатель будет положительным. Для точек из остальных секторов показатели будут отрицательными. Большая часть показателей для точек будет положительной. Теперь сложим все такие показатели и усредним (добавляем -1 в знаменатель, это связано с числом степеней свободы):

$$\frac{\sum (x_i - \overline{X}) \cdot (y_i - \overline{Y})}{N - 1}$$

```
sum(
    (wh_data$weight - mean(wh_data$weight)) *
        (wh_data$height - mean(wh_data$height))
) / (length(wh_data$weight) - 1)
```

```
## [1] 61.90428
```

Такой показатель называется ковариацией.

```
cov(wh_data$weight, wh_data$height)
```

```
## [1] 61.90428
```

Таким образом мы расчитали количественный показатель силы и направления взаимосвязи двух переменных. Чем больше значение ковариации, тем сильнее взаимосвязь и если она положительная, то и корреляция положительная.

Теперь введем именно показатель корреляции:

$$r_{xy} = \frac{cov(x, y)}{\sigma_x \sigma_y} \tag{3.1}$$

Таким образом все значения показателя корреляции всегда лежат в интервале [-1; 1].

```
cov(wh_data$weight, wh_data$height) /
    (sd(wh_data$weight) * sd(wh_data$height))
```

```
## [1] 0.6630431
```

или

```
cor(wh_data$weight, wh_data$height)
```

```
## [1] 0.6630431
```

Теперь укажем более традиционную формулу для корреляции (его так же называют коэффициентом корреляции Пирсона):

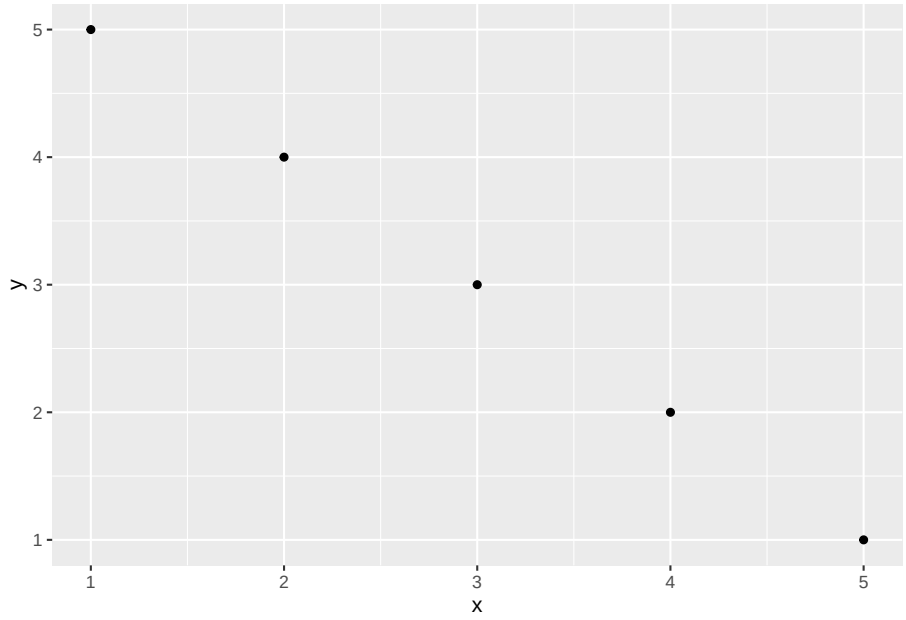
$$r_{xy} = \frac{\sum (x_i - \overline{X}) \cdot (y_i - \overline{Y})}{\sqrt{\sum (x_i - \overline{X})^2 (y_i - \overline{Y})^2}} \tag{3.2}$$

Квадрат коэффициента корреляции называется коэффициентом детерминации и показывает, в какой степени дисперсия одной переменной обусловлена влиянием другой переменной. Принимает значения [0,1]. Теперь перейдем к вопросу, как проверять статистические гипотезы используя коэффициент корреляции Пирсона. возвращаемся к примеру с ростом и весом. Сформулируем нулевую гипотезу: $r_{xy} = 0$. Альтернативная гипотеза говорит: $r_{xy} \neq 0$. В нашем эксперименте мы получили определенную связь между ростом и весом. Теперь надо найти p-уровень значимости для гипотезы. Мы будем расчитывать t-критерий при числе степеней свободы $N - 2$ (на вопрос почему ответим позже в разделе про линейную регрессию). Еще один пример: Чему равен коэффициент корреляции в данной выборке

x	y
4	2
5	1
2	4
3	3
1	5

Решение:

```
tibble(
    x = c(4, 5, 2, 3, 1),
    y = c(2, 1, 4, 3, 5)
) |>
    ggplot(aes(x, y)) +
    geom_point()
```




```
cor(c(4, 5, 2, 3, 1), c(2, 1, 4, 3, 5))
```

```
## [1] -1
```

3.2 Условия приенения коэффициента корреляции

Завершая разговор о корреляциях остановимся на нескольких важных моментах, посвященных правильной интерпретации получаемых данных в корреляционных исследованиях.

Первая из них - ошибка корреляции. Её суть заключается в том, что положительная или отрицательная взаимосвязь еще не обязательно говорит о причинно-следственной зависимости. Допустим мы решили выяснить, действительно ли компьютерные игры негативно влияют на подростков (агрессивное поведение). Взяли 500 школьников и посмотрели как часто они дерутся со сверстниками и как часто они играют в игры. Если мы получили значимую положительную корреляцию, это еще не значит, что именно игры стали причиной агрессивного поведения. Возможно это агрессивные школьники любят играть в компьютерные игры.

Сама по себе корреляция не означает наличие причинно-следственной зависимости, но может её означать. Корреляция может подтверждать выдвинутую теорию.

Второй момент - это влияние так называемой третьей переменной. Это такая неявная перменная, которая не входит в рассмотрение, но именно она является причиной корреляции. Например: размер ноги школьника хорошо коррелирует со знаниями математики. Потому что чем он старше, тем у него и размер ноги больше, и знаний больше.

3.3 Регрессия с одной независимой перменной

В этом и следующих уроках мы научимся работать с одномерным регрессионным анализом, который позволяет проверять гипотезы о взаимосвязи одной количественной зависимой переменной и нескольких независимых.

Сначала мы познакомимся с самым простым вариантом - простой линейной регрессией, при помощи которой можно исследовать взаимосвязь двух переменных. Затем перейдем к множественной регрессии с несколькими независимыми переменными.

Регрессионный анализ это набор методов, позволяющих исследовать взаимосвязь различных переменных между собой. Начнем мы с простой линейной регрессии, которая как и коэффициент корреляции позволяет нам исследовать взаимосвязь двух количественных переменных. Однако позволяет делать это более интересным образом.

3.3.1 Линия регрессии

Одномерный регрессионный анализ применяется для исследования взаимосвязи двух переменных. Пусть в нашем случае это буду x и y. Переменная по оси y называется **зависимая переменная**, а x - это **независимая переменная** или **предиктор**.

Регрессионный анализ обычно применяется, чтобы посмотреть как одна переменная объясняет другую переменную. Например, хотим посмотреть как на качество товара влияет его цена. Если две переменные как-то взаимосвязаны между собой удобно добавить линию на диаграмму. Нам нужно, чтобы она следовала за облаком точек на диаграмме и чтобы она хорошо описывала распределение данных.

Мы знаем, что любая линия задается двумя параметрами:

$$y = b_0 + b_1 x$$

здесь, b_0 (intercept) отвечает за точку пересечения линии с осью Y, а b_1(slope) за наклон линии. Один из самых простых методов построения такой прямой это метод наименьших квадратов.

3.3.2 Метод наименьших квадратов

Определение 3.3.1 *МНК - метод нахождения оптимальных параметров линейной регресии, таких, что сумма квадратов ошибок (остатков) была минимальна.*

Остаток, это расстояние от отдельно выбранной точки до прямой. Оно рассчитывается как:

$$e_i = y_i - \hat{y}_i$$

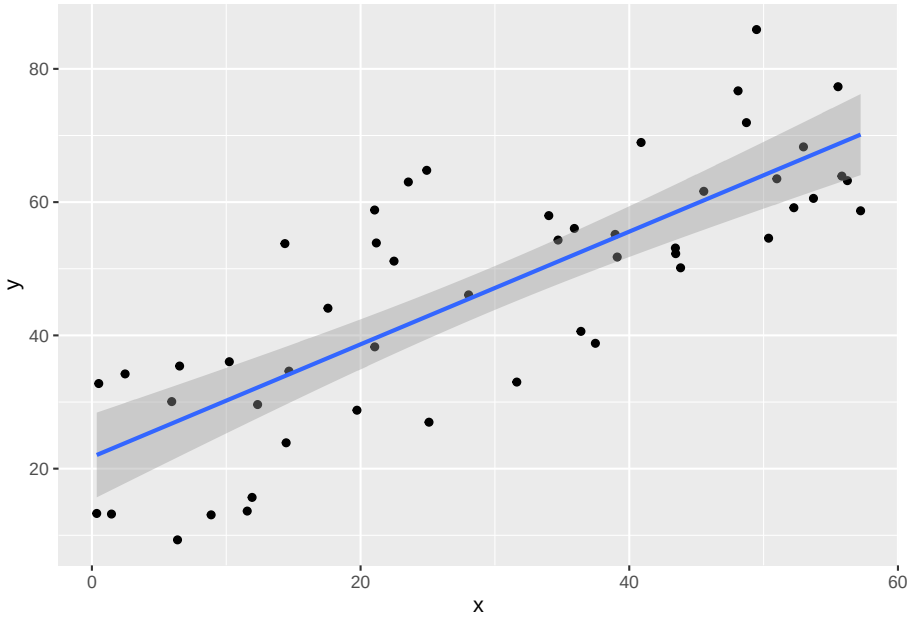
Сложением квадратов мы избегаем зануление одинаково отстоящих точек. Расчет:

$$b_1 = \frac{sd_y}{sd_x} r_{xy}, b_0 = \overline{y} - b_1 \overline{x}$$

```
data <- tibble(
    x = runif(50, 0, 60),
    y = runif(50, 0, 40)
) |>
    mutate(
        y = y + x
    )
lm(y ~ x, data) |>
    tidy() |>
    kable()
```

term	estimate	std.error	statistic	p.value
(Intercept)	21.7488657	3.1916662	6.814267	0
x	0.8456498	0.0923671	9.155319	0

```
data |> ggplot(aes(x, y)) +
    geom_point() +
    geom_smooth(method = "lm", formula = y ~ x)
```



Задача: На графике изображена зависимость двух количественных переменных X и Y . Рассчитайте коэффициент b1 для регрессионной прямой, если коэффициент детерминации равен 0.25:

$$M_x = 15(\text{выборочное среднее}), D_x = 25, M_y = 10, D_y = 36$$

Формула для b_1: $b_1 = \frac{sd_y}{sd_x} r_{xy}$. $sd = \sqrt{D}$, коэффициент детерминации: r^2_{xy}. Итого:

```
(sqrt(36) / sqrt(25)) * sqrt(0.25)
```

```
## [1] 0.6
```

3.3.3 Гипотеза о значимости взаимосвязи и коэффициент детерминации

Теперь, когда мы построили линию регрессии мы хотим ответить на вопрос «на сколько статистически значима взаимосвязь двух наших переменных?». За взаимосвязь отвечает именно b_1. Если коэффициент корреляции 0, то прямая проходит параллельно оси X. Таким образом, если переменные не связаны между собой, то b_1 становися равен 0.

То есть появляются две гипотезы:

$$H_0 : b_1 = 0$$

$$H_1 : b_1 \neq 0$$

Теперь появляется t-критерий, который говорит, что если бы мы многократно повторяли эксперимент и данные на самом деле не зависят, то выборочные значения коэффициентов b_1 распределились бы нормально относительно 0 и отклонялись бы то в правую, то в левую сторону.

Теперь можно расчитать t-критерий который будет сообщать насколько выборочное b_1 отклонится от ожидаемого.

И здесь он будет соответствовать формуле: $t = \frac{b_1}{se}$.

3.3.4 Коэффициент детермнации

Одномерный дисперсионный анализ позволяет нам построить некоторую модель взаимосвязи двух количественных переменных, проверить гипотезу о наличии взаимосвязи, рассчитав соответствующий p-уровень значимости и получить значение коэффициента детерминации (R^2). Мы уже немного говорили о том, что это такое, когда обсуждали коэффициент корреляции. Мы выяснили, что R^2 это часть дисперсии одной переменной, обусловленная другой переменной.

В контексте регрессионного анализа, это:

Определение 3.3.2 *R^2 – это доля дисперсии зависимой переменной (Y), объясняемая регрессионной моделью.*

Допустим, мы решили выяснить, как качество определяется ценой товара и построили регрессионную модель. Для этого считаем:

$$R^2 = 1 - \frac{SS_{res}}{SS_{total}} \tag{3.3}$$

SS_{res} – сумма квадратов расстояний от наблюдений до регрессионной прямой. SS_{total} – сумма квадратов расстояний от наблюдения до среднего значения.

Если значение R^2 велико, это говорит, что почти вся изменчивость зависимой переменной обуславливается её взаимосвязью с независимой переменной. То есть регрессионная модель очень хорошо объясняет и описывает поведение и изменчивость зависимой переменной. Чем больше R^2, тем лучше модель справляется с поставленной задачей – объясняет и предсказывает значения зависимой переменной.

3.4 Условия применения линейной регрессии с одним предиктором.

Подведем итоги. Допустим у нас есть две количественные переменные и мы хотим узнать как они связаны между собой. Более того, мы не просто хотим знать их связь, но и как одна переменная влияет на поведение другой.

Например мы хотим узнать, как уровень безработицы влияет на процент преступности. Или как средний бал ЕГЭ у абитуриента определяет его успеваемость на первом курсе университета.

Таким образом мы хотим посмотреть как взаимосвязаны две переменные между собой. Можем провести регрессионный анализ и посмотреть как линия регрессии максимально хорошо описывает эту взаимосвязь, построить уравнение регрессии, проверить гипотезу о статистически значимой взаимосвязи между ними и посчитать какой процент изменчивости зависимой переменной обуславливается нашей моделью. Все это вместе делает этот метод незаменимым для решения колоссального числа задач.

Перед тем как посмотреть, как регресионный анализ позволяет нам исследовать реальные данные, поговорим про некоторые ограничения этого метода, какие требования к данным возникают когда мы хотим использовать его.

3.4.1 Условия применения

Условия выглядят следующим образом: * Линейная зависимость X и Y * Нормальное распределение остатков * Гомоскедастичность – постоянная изменчивость остатков на всех уровнях независимой переменной.

Чтобы познакомиться с этими требованиями наглядно, рассмотрим следующие примеры. Код генерации примеров:

```
# Based on code at
# https://github.com/ShinyEd/intro-stats/blob/master/slr_diag/app.R
make_example <- function(type) {
    make_data <- function(type) {
        n <- 250
        if (type == "linear.up") {
            x <- c(runif(n - 2, 0, 4), 2, 2.1)
            y <- 2 * x + rnorm(n, sd = 2)
        }
        if (type == "linear.down") {
            x <- c(runif(n - 2, 0, 4), 2, 2.1)
            y <- -2 * x + rnorm(n, sd = 2)
        }
```

```
        if (type == "curved.up") {
            x <- c(runif(n - 2, 0, 4), 2, 2.1)
            y <- 2 * x^4 + rnorm(n, sd = 16)
        }
        if (type == "curved.down") {
            x <- c(runif(n - 2, 0, 4), 2, 2.1)
            y <- -2 * x^3 + rnorm(n, sd = 9)
        }
        if (type == "fan.shaped") {
            x <- seq(0, 3.99, 4 / n)
            y <- c(
                rnorm(n / 8, 3, 1),
                rnorm(n / 8, 3.5, 2),
                rnorm(n / 8, 4, 2.5),
                rnorm(n / 8, 4.5, 3),
                rnorm(n / 4, 5, 4),
                rnorm((n / 4) + 2, 6, 5))
        }
        tibble(x = x, y = y)
    }

    my_data <- make_data(type)
    lm_results <- lm(y ~ x, data = my_data)
    xcon <- seq(min(my_data$x), max(my_data$x), 0.025)
    predictor <- tibble(x = xcon)
    pred_inter <- as.data.frame(
        predict(lm_results, newdata = predictor, interval = "prediction")
    ) |> mutate(x = xcon)

    r_squared <- round(summary(lm_results)$r.squared, 4)
    corr_coef <- round(sqrt(r_squared), 4)

    regression_plot <- my_data |>
        modelr::add_predictions(lm_results, var = "y_end")|>
        ggplot(aes(x = x, y = y)) +
            geom_point() +
            geom_smooth(method = lm, formula = y ~ x) +
            geom_segment(aes(xend = x, yend = y_end, color = "red")) +
            geom_ribbon(data = pred_inter,
                aes(x = x, y = fit, ymin = lwr, ymax = upr),
                alpha = 0.2
            ) +
            labs(
                title = paste0(
                    "Regression Model\n",
```

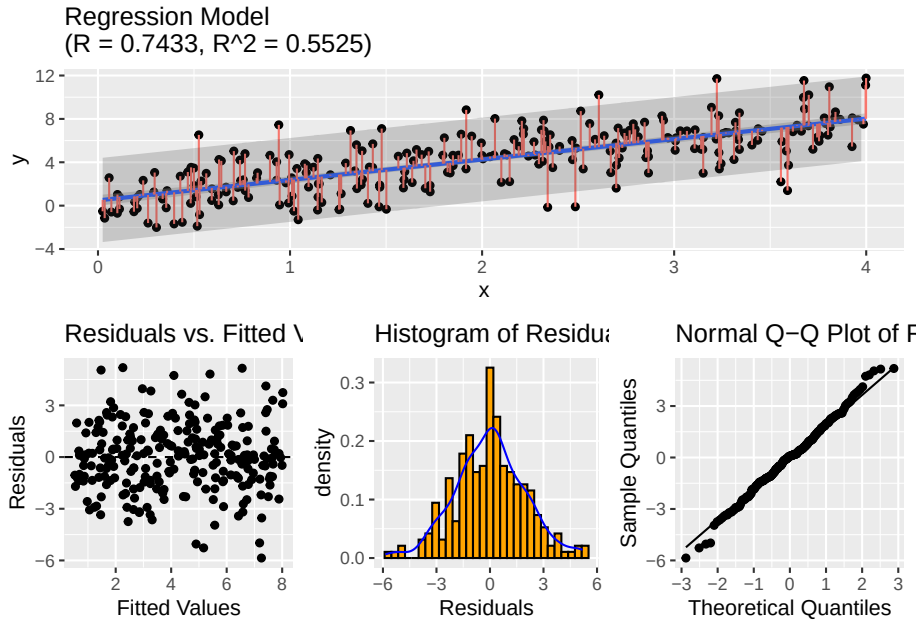
```
                    "(R = ", corr_coef, ", ",
                    "R^2 = ", r_squared, ")")
            ) + theme(legend.position = "none")
    residuals <- summary(lm_results)$residuals
    predicted <- predict(lm_results, newdata = data.frame(x = my_data$x))
    residual_plot <- tibble(x = predicted, y = residuals) |>
        ggplot(aes(x, y)) +
            geom_point() +
            geom_hline(yintercept = 0, linetype = 2) +
            labs(
                title = "Residuals vs. Fitted Values",
                x = "Fitted Values",
                y = "Residuals"
            )
    resid_hist <- tibble(res = residuals) |>
        ggplot(aes(x = res)) +
        geom_histogram(aes(y = after_stat(density)),
            bins = 30,
            fill = "orange", color = "black") +
        geom_density(kernel = "gaussian", color = "blue") +
        labs(
            title = "Histogram of Residuals",
            x = "Residuals"
        )
    qq_plot <- tibble(res = residuals) |>
        ggplot(aes(sample = res)) +
        geom_qq() + geom_qq_line() +
        labs(
            title = "Normal Q-Q Plot of Residuals",
            x = "Theoretical Quantiles",
            y = "Sample Quantiles"
        )

    grid.arrange(
        regression_plot,
        arrangeGrob(residual_plot, resid_hist, qq_plot,
            ncol = 3, nrow = 1),
        nrow = 2)
}
```

3.4.1.1 Пример 1

Пусть у нас линейная положительная взаимосвязь двух переменных.

```
make_example("linear.up")
```

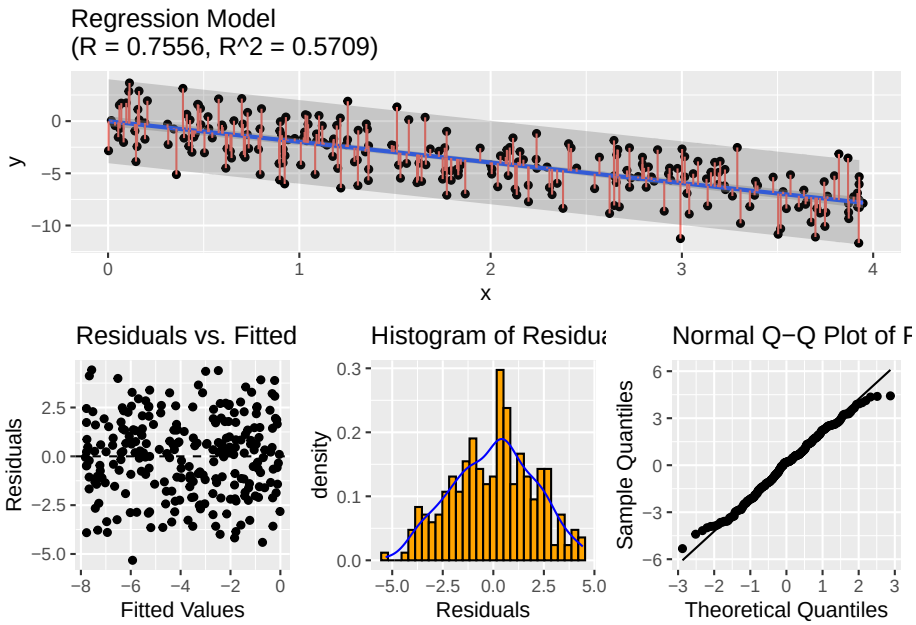


Видим, что наши данные распределены линейно и у нас видно положительные корреляции. Для каждой точки мы расчитываем остатки (красные линии от точек). По графику зависимости остатков (левый нижний график), видим, что остатки нормально распределились вокруг регрессионной линии. То есть половина остатков положительная, половина отрицательная. Если мы построим гистограмму распраделения остатков, увидим, что это нормальное распределение. Дополнительно проверяем это через QQ-plot. Таким образом у нас соблюдены требования нормальности распределения остатков и линейности взаимосвязи.

3.4.1.2 Пример 2

Теперь посмотрим, что произойдет, если взаимосвязь будет линейной, но отрицательной.

```
make_example("linear.down")
```

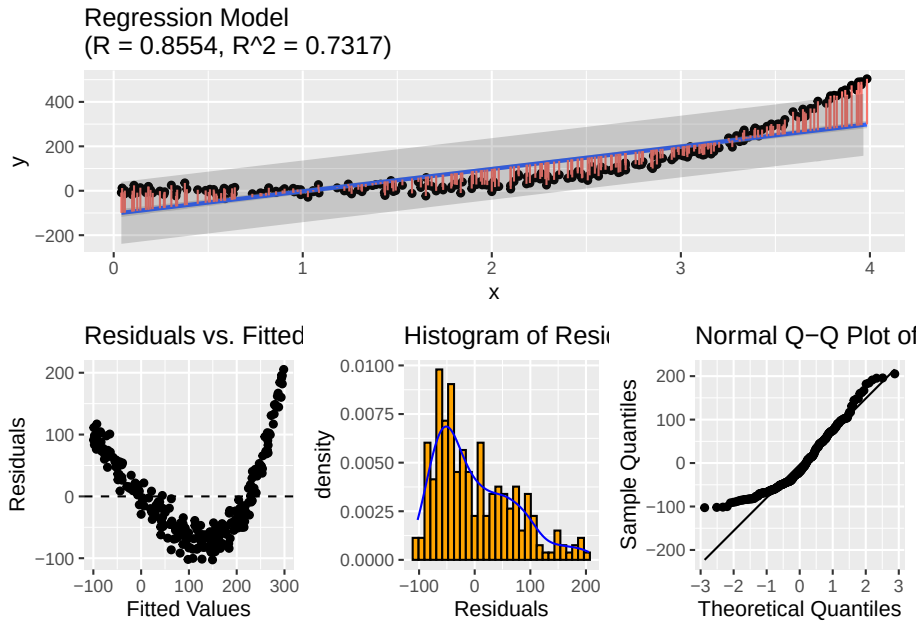


Коэффициент корреляции стал отрицательным (R). Видим, что остатки так же распределены нормально. Таким образом, и в этом случае выполняются требования к линейной регрессии.

3.4.1.3 Пример 3

Следующий пример будет интереснее: искревленная зависимость.

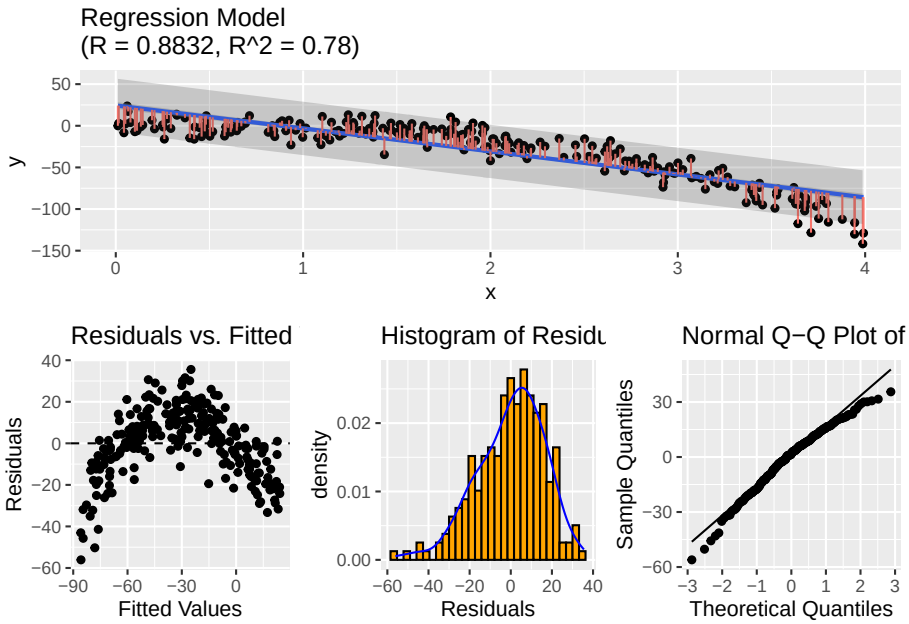
```
make_example("curved.up")
```



Зависимость явно нелинейная, но положительная. Коэффициент регрессии здесь высокий, однако применять линейную регрессию здесь не совсем оправдано. Если мы построим гистограмму и qq-plot остатков, то увидим, что их распределение не является нормальным.

3.4.1.4 Пример 4

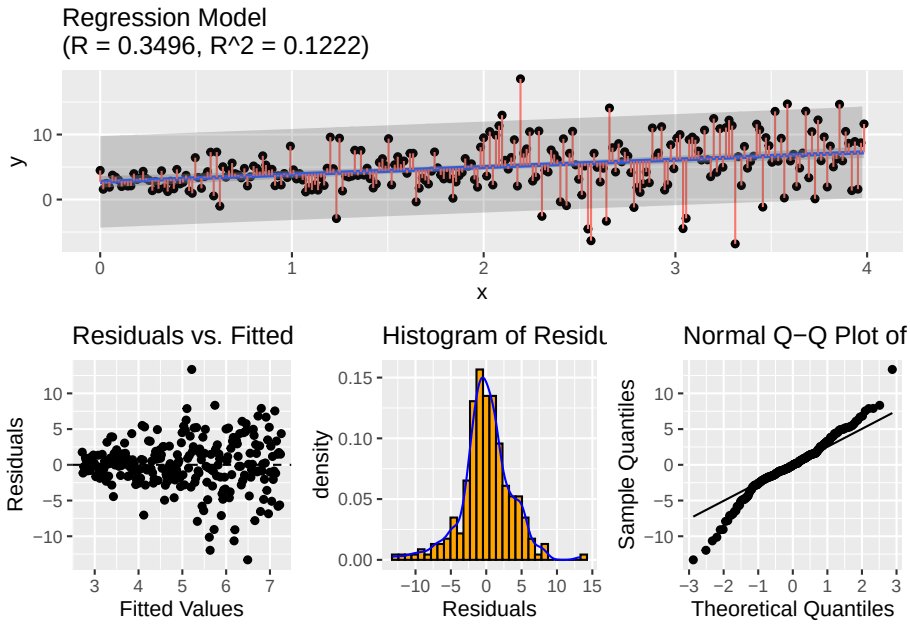
```
make_example("curved.down")
```



В этом примере мы так же наблюдаем ассиметричное распределение остатков. На графике расброса точек (верхний) может показаться, что все обстоит не так плохо, но прогноз регрессии будет в таких случаях сильно расходиться с реальными данными.

3.4.1.5 Пример 5

```
make_example("fan.shaped")
```



В этом примере зависимость в целом линейна. Но так же видно, что чем дальше мы двигаемся по оси x, тем больше становятся остатки. Мы видим, что остатки будут рассеиваться. В этом случае линейная регрессия так же будет работать не совсем корректно.

Представленные в этих примерах графики являются хорошим примером анализа применимости линейной регрессии. Если все условия выполняются, то линейная регрессия будет хорошо описывать зависимость.

Далее мы перейдем к решению реальных задач на линейную регрессию.

3.5 Применение регрессионного анализа и интерпретация результатов

В качестве датафрейма мы будем использовать набор данных различных экономических показателей для каждого штата США (файл states.csv).

```
states <- read_csv("states.csv", show_col_types = FALSE)
```

Table 3.3: Descriptive statistics

state	metro_res	white	hs_grad	poverty	female_house
Alabama	55.4	71.3	79.9	14.6	14.2

state	metro_res	white	hs_grad	poverty	female_house
Alaska	65.6	70.8	90.6	8.3	10.8
Arizona	88.2	87.7	83.8	13.3	11.1
Arkansas	52.5	81.0	80.9	18.0	12.1
California	94.4	77.5	81.1	12.8	12.6
Colorado	84.5	90.2	88.7	9.4	9.6

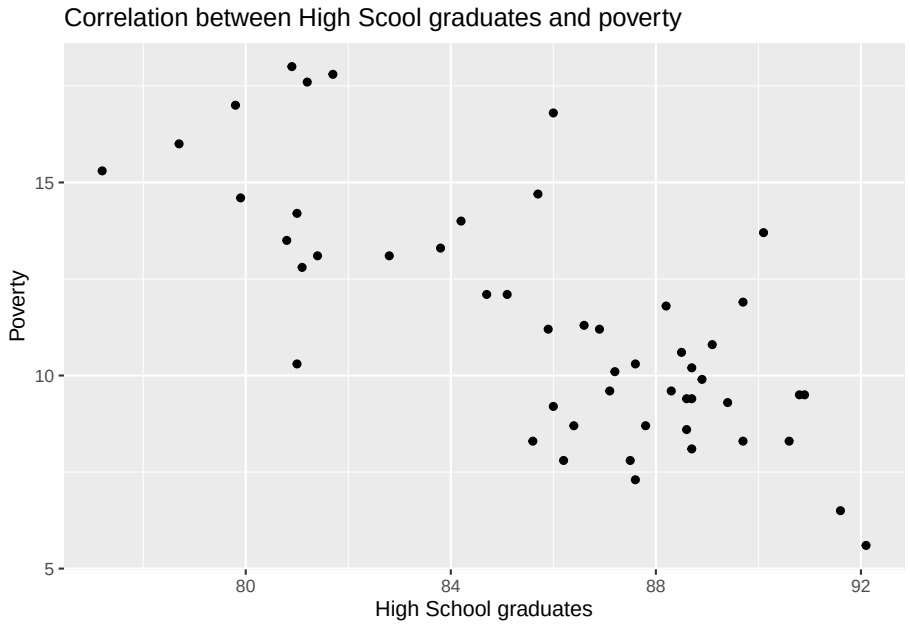
Statistic	N	Mean	SD	Min	Max
female_house	51	11.63333	2.356155	7.8	18.9
hs_grad	51	86.01176	3.725998	77.2	92.1
metro_res	51	72.24902	15.275894	38.2	100.0
poverty	51	11.34902	3.099185	5.6	18.0
white	51	81.71961	13.897223	25.9	97.1

Посмотрим на переменные, которые представлены в этом датафрейме.

- metro_res (metropolitan residence) говорит нам какой процент населения штата живет в столичной области.
- white - процент белокожего населения штата
- hs_grad - процент людей, имеющих среднее образование
- poverty - процент людей, проживающих в бедности
- female_house - процент семей, где женщина является домохозяйкой

Первая наша задача будет применить линейную регрессию с одной зависимой переменной и исследуем, как взаимосвязаны бедность и уровнь образования.

Для начала имеет смысл просто построить диаграмму рассеяния чтобы составить первое впечателение о характере взаимосвязи наших переменных.



Очевидно, что за некоторыми исключениями, характер нашей зависимости ялвяется линейным. При этом можно заметить, что взаимосвязь явно отрицательная.

```
cor(states$hs_grad, states$poverty)
```

```
## [1] -0.7468583
```

Линейность взаимосвязи позволяет нам применить регрессионный анализ. Анализ остатков проведем чуть позже. Сформулируем задачи и гипотезы, которые мы будем проверять при помощи регрессионного анализа.

В качестве зависимой переменной будет выступать показатель бедности, в качестве независимой – уровень образования.

Первое, что мы сделаем, это построим регрессионную модель, которая максимально удачным образом объясняет взаимосвязь двух наших перемененных. В нашем случае это будет уравнение регрессионной прямой.

Далее мы будем выяснять насколько хорошо эта модель будет описывать поведение наших данных. Для этого мы будем считать коэффициент детерминации.

Далее мы ответим на вопрос, отличается ли коэффициент при независимой переменной в линейной модели от 0. Это и будет нашей нулевой гипотезой так как если бы никакой взаимосвязи не было, то наша линия регрессии

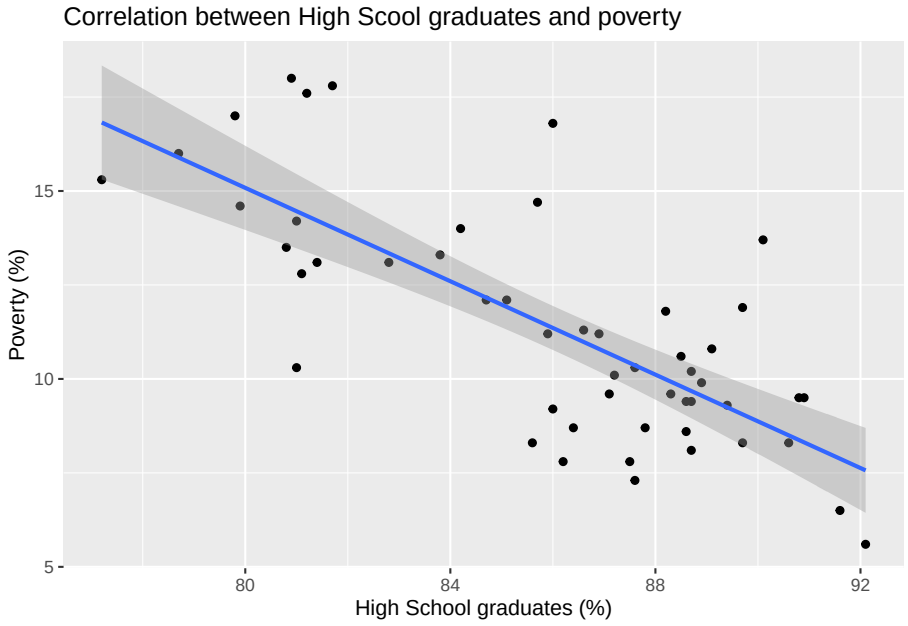
была бы параллельно оси X. Если этот коэффициент не равен 0, значит возникает какой-то наклон, который говорит нам о взаимосвязи.

Далее мы основываясь на независимой переменной можно предсказать чему будет равна зависимая.

3.5.1 Анализ

Зная коэффициент корреляции и описательную статистику, мы можем рассчитать коэффициенты регрессии, коэффициент детерминации и проверить наши статистические гипотезы при помощи T-теста.

```
ggplot(states, aes(hs_grad, poverty)) +
    geom_point() +
    geom_smooth(method = "lm", formula = y ~ x) +
    labs(
        title = "Correlation between High Scool graduates and poverty",
        x = "High School graduates (%)",
        y = "Poverty (%)"
    )
```




```
states_lm <- lm(poverty ~ hs_grad, states)
states_lm |> tidy() |> kable()
```

term	estimate	std.error	statistic	p.value
(Intercept)	64.7809658	6.8025952	9.522978	0
hs_grad	-0.6212167	0.0790165	-7.861864	0

```
print(summary(states_lm))
```

```
##
## Call:
## lm(formula = poverty ~ hs_grad, data = states)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1624 -1.2593 -0.2184  0.9611  5.4437
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 64.78097    6.80260   9.523 9.94e-13 ***
## hs_grad     -0.62122    0.07902  -7.862 3.11e-10 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.082 on 49 degrees of freedom
## Multiple R-squared:  0.5578, Adjusted R-squared:  0.5488
## F-statistic: 61.81 on 1 and 49 DF,  p-value: 3.109e-10
```

Мы обнаружили статистически значимую взаимосвязь наших переменных ($p < 0.05$). Напротив Intercept так же есть значение p-value. Тут проверяется гипотеза, что Intercept отличен от 0. Это не позволяет сделать такого сильного вывода, как во втором случае, но так же дает интересную информацию. Если предположить существование такого штата, где процент людей с высшим образованием равен 0, то мы ожидаем, что в таком штате будет примерно 64.78% людей проживающих за чертой бедности.

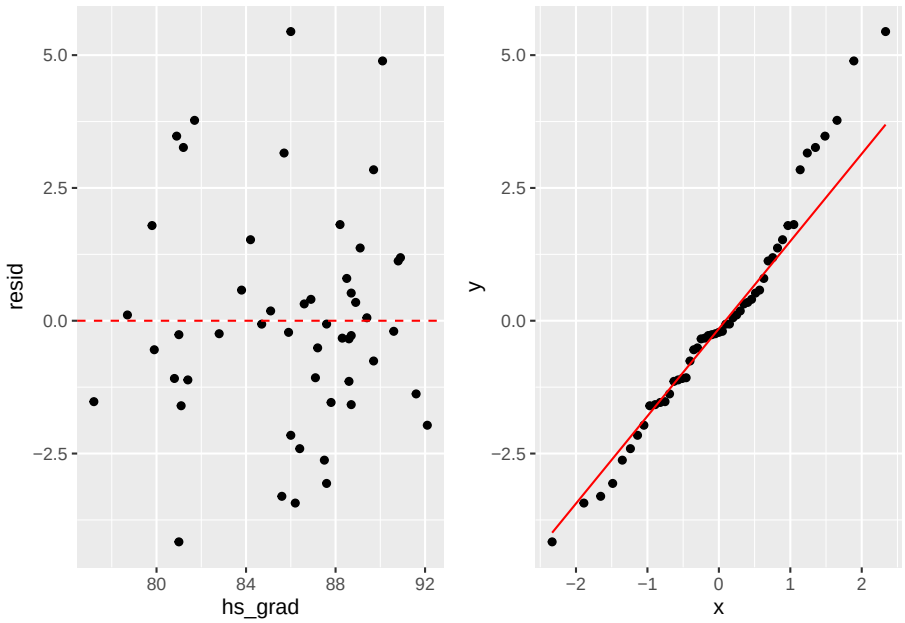
Удостоверившись в том, что взаимосвязь между нашими переменными статистически значимая, интерпретируем значение коэфициента наклона. Значение -0.62 говорит о том, что взаимосвязь отрицательная и означает, что с каждым процентом, увеличения людей, имеющих высшее образование, мы ожидаем, что количество людей, проживающих в бедности будет уменьшаться на 0.62%. Поэтому в случае применения регрессионного анализа в таких задачах, коэффициенты перед независимыми переменными имеют определенный смысл: как будет изменяться значение зависимой переменной от независимой.

Далее, значение коэффициента детерминации 0.56 означает, что практически 56% изменчивости нашей зависимой переменной объясняется нашей моделью. В целом это не так уж и много – почти половина изменчивости обуславливается не включенными в модель факторами. У коэффициента детерминации так же есть свой p-уровень значимости и F-значение, полученное при применении дисперсионного анализа при проверки гипотезы о том, что модель позволяет объяснить поведение нашей зависимой переменной. Как интерпретировать уровнь значимости для всей модели станет понятнее, когда мы разберемся с множественно регрессией, когда одновременно используется несколько предикторов.

При помощи регрессионного анализа, мы узнали как взаимосвязаны две переменные. Мы посмотрели насколько значима эта взаимосвязь, она оказалось отрицательной, построили линейную модель, выяснили какой процент дисперсии обуславливается взаимосвязью с другой переменной и узнали какие ожидать изменение уровня бедности с единичным изменением уровня образования.

Теперь посмотрим на остатки, получившейся модели.

```
states_resid <- states |>
    add_residuals(states_lm)
grid.arrange(
    ggplot(states_resid, aes(hs_grad, resid)) +
        geom_point() +
        geom_hline(yintercept = 0, linetype = 2, color = "red"),
    ggplot(states_resid, aes(sample = resid)) +
        geom_qq() +
        geom_qq_line(color = "red"),
    nrow = 1, ncol = 2
)
```



На графиках изображено распределение остатков: Левый график: распределение остатков на различных уровнях независимой переменной Правый график: qq - plot для остатков. Основываясь на этих данных, можно заключить, что в целом требования гомоскедастичности и нормальности распределения остатков выполнено (хотя и присутствуют заметные положительные отклонения от нуля).

Таким образом, делаем вывод, что применение регрессионного анализа в данном случае было обоснованным и мы можем доверять нашей модели. Следовательно мы можем попробовать предсказать дальнейшие значения.

3.6 Задача предсказания значений зависимой переменной

Регрессионную прямую еще иногда называют линией тренда. По ней мы можем предсказать, чему будут равны значения зависимой переменной на определенном уровне значения независимой переменной.

Например, если бы мы решили выяснить, чему равняется процент людей, проживающих за чертой бедности в штате, где среднее образование имеют 95% населения. В нашей выборке (она здесь совпадает с генеральной совокупностью) такого наблюдения не было. Но основываясь на регрессионном анализе, мы можем сделать предсказание относительно такого события.

```
tibble(hs_grad = c(95)) |>
    add_predictions(states_lm) |>
    kable()
```

hs_grad	pred
95	5.765378

Таким образом, основываясь на регрессионном анализе мы можем предсказывать несуществующие значения. Этот метод используется повсеместно. Однако эти предсказания имеют некоторые ограничения. Если мы вернемся к примеру с бедностью, нет ничего удивительного в том, что при некоторых значениях уровня образования, предсказанное значение будет принимать отрицательные значения, что совершенно бессмысленно.

Другая частая ситуация – предсказание работает только на каком-то промежутке значений. Например, если мы будем изучать зависимость роста от возраста, то до 20 лет мы будем получать линейную взаимосвязь, но потом значение роста перестанет меняться.

Однако, в случае с уровнем бедности и образованности, мы можем объяснить установленную взаимосвязь двумя способами:

1. Чем лучше люди учатся, тем лучше они работают и тем больше их благосостояние.
2. Чем меньше благосостояние населения, тем меньше у них времени на учебу.

Сам факт взаимосвязи совершенно ничего не говорит о порядке причинно-следственных связей.

3.7 Регрессионный анализ с несколькими независимыми переменными

3.7.1 Множественная регрессия

Множественная регрессия позволяет исследовать влияние сразу нескольких независимых переменных на одну зависимую переменную.

Мы выяснили, как влияет уровень образования на уровень бедности. В большинстве случаев используется множетсво различных факторов. Мы можем так же узнать, как процент белого населения влияет на показатель

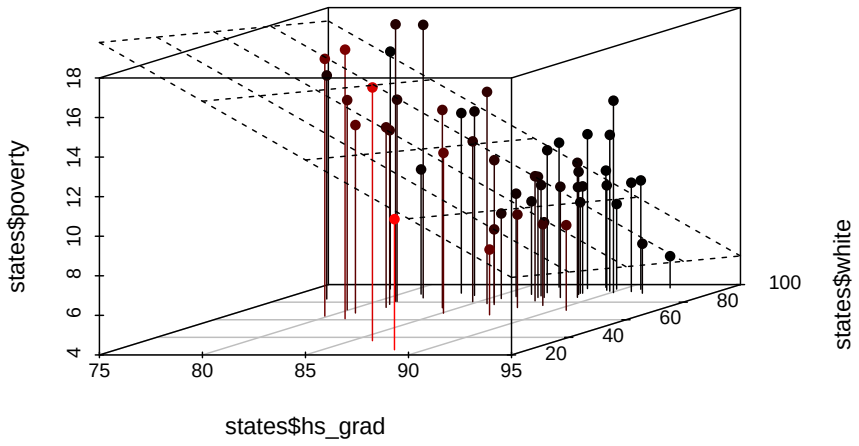
зависимой переменной, то мы можем включить эту переменную в нашу модель. Тогда уравнение для нашей модели примет несколько другой вид:

$$\hat{y} = b_0 + b_1 x_1 + b_2 x_2$$

Если мы хотим включать произвольное число переменных, то модель принимает следующий вид:

$$\hat{y} = b_0 + \sum_{i=1}^{N} b_i x_i \tag{3.4}$$

В случае нашей задачи, можно визуализировать зависимость следующим



образом:
Кроме того, теперь регрессия является не прямой линией, а плоскостью. Как и раньше, с увеличением числа людей с средним образованием, уровень бедности падает. Но при этом с ростом числа белого населения, уровень бедности так же начинает немного снижаться.

Таким образом видим, что и уровень образования и уровень белого населения отрицательно взаимосвязан с зависимой переменной. Поэтому в задачах с множественной регрессией мы так же будет оценивать значения коэффициентов.

3.7.2 Требования к данным

- Линейная зависимость переменных

- Нормальное распределение остатков
- Гетероскедастичность
- Проверка на мультиколлинеарность
- Нормальное распределение переменных (желательно)

Построим регрессионную модель, которая включает сразу все параметры из датасета.

```
summary(lm(
    poverty ~ metro_res + white + hs_grad + female_house,
    data = states))
```

```
##
## Call:
## lm(formula = poverty ~ metro_res + white + hs_grad + female_house,
##     data = states)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.289 -1.506 -0.323  1.235  4.449
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  66.47653   12.58990   5.280 3.41e-06 ***
## metro_res    -0.05632    0.01955  -2.881    0.006 **
## white        -0.04814    0.03306  -1.456    0.152
## hs_grad      -0.55471    0.10491  -5.288 3.33e-06 ***
## female_house  0.05054    0.24330   0.208    0.836
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.934 on 46 degrees of freedom
## Multiple R-squared:  0.6416, Adjusted R-squared:  0.6104
## F-statistic: 20.58 on 4 and 46 DF,  p-value: 8.884e-10
```

Теперь у нас есть значения Intercept и коэффициентов перед каждой из переменной. Физический смысл Intercept в целом сохраняется. При помощи T критерия мы оцениваем отличие параметра от 0. Если рассматривать значение каждого коэффициента по отдельности, то можно сказать следующее.

- **metro_res** – статистически значимо взаимосвязан с зависимой переменной. Он отрицательный, значит чем больше людей живет

в столичной области, тем меньше уровнь бедности. Если бы мы допустили, что все остальные перменные зафиксированы, то мы бы увидели, что с каждым увеличением процента людей, проживающих в столичной области, уровень бедности снижался бы на 0.06%.
- **white** – этот параметр судя по p-value значимо не отличается от 0, поэтому мы ничего не можем сказать о его значимости.
- **hs_grad** – Видим, что он самый большой по модулю и что у него самый низкий p.value, что говорит нам о том, что этот коэффициент вносит существенный вклад в нашу модель.
- **female_house** – аналогично white.

Кроме того видим, что у нас есть параметр adjusted r-squared. Это скорректированный коэффициент детерминации. Рассчитывается он при включении в модель дополнительных независимых переменных.

3.8 Выбор наилучшей модели

Подведем итог. Мы решили для начала выяснить как на нашу зависимую переменную (процент бедности) влияет набор социально-экономических показателей и сначала построили простую модель с двумя переменными. Получилось не так плохо: мы смогли найти значимую взаимосвязь между этими переменными и построить регрессионную модель. Посчитали какой процент изменчивости обуславливается этой моделью.

Затем мы включили все показатели в модель. Это привело к некоторому улучшению. Мы смогли поймать еще одну переменную, которая так же имеет статистически значимую взаимосвязь с нашей переменной (процент людей, проживающих в столичном регионе).

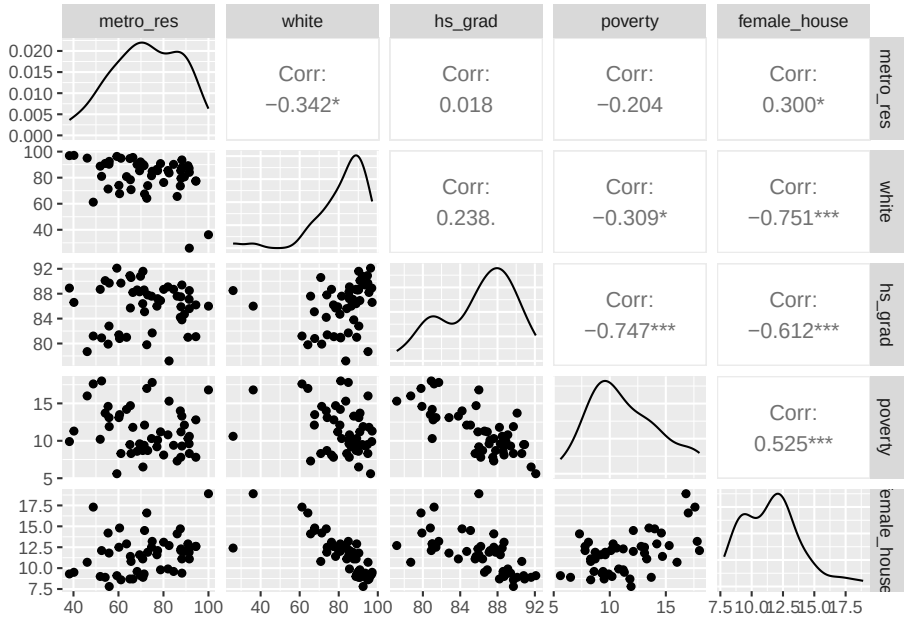
Однако полученная нами модель далеко не самая лучшая из доступных нам. Здесь надо вспомнить про требование мультиколлинеарности. Мультиколлинеарность означает очень сильную взаимосвязь между какими-то из независимых переменных.

Включение таких переменных в модель нецелесообразно: если две независимые переменные между собой сильно взаимосвязаны, то достаточно только одной из них, чтобы хорошо объяснить зависимость. С точки зрения математики, если у нас сильная корреляция между предикторами, то становится проблематично рассчитать все показатели.

То есть, при построении регрессонной модели, далеко не факт, что включение всех переменных в регрессионный анализ приведет к получению наилучшего результата (большего R^2).

Посмотрим как можно подобрать оптимальную модель.

```
ggpairs(states |> select(-c(state)))
```



Видим знакомую уже нам сильную корреляцию между poverty и hs_grad. Кроме того видим, что переменная female_house хорошо коррелирует со всеми переменными. Не исключено, что наличие этой переменной в нашей модели ухудшает её работу.

Один из возможных методов улучшения модели выглядит следующим образом. Мы строим модель в которую включаем все переменные и считаем Adj. R-squared. Далее мы будем по очереди удалять каждую переменную.

```
cols <- colnames(states) |>
    purrr::discard(\(x) x %in% c("state", "poverty"))
cols_to_del <- cols |> append("", after = 0)
model_selection <- tibble(to_del = cols_to_del) |>
    mutate(
        model = str_c(
            "poverty ~ ",
            map(to_del,
                \(y) str_flatten(
                    purrr::discard(
                        cols,
                        \(x) x == y),
                    collapse = " + ")
```

```
                )
            ),
        adj_r_sq = map(
            model,
            \(x) round(summary(
                lm(formula(x), data = states)
            )$adj.r.squared, digits = 2))
    )
kable(model_selection)
```

to_del	model	adj_r_sq
	poverty ~ metro_res + white + hs_grad + female_house	0.61
metro_res	poverty ~ white + hs_grad + female_house	0.55
white	poverty ~ metro_res + hs_grad + female_house	0.6
hs_grad	poverty ~ metro_res + white + female_house	0.39
female_house	poverty ~ metro_res + white + hs_grad	0.62

Видим, что оптимальная модель возникает при удалении female_house. Продолжаем удалять переменные предварительно выкинув переменную female_house.

```
cols <- colnames(states) |>
    purrr::discard(\(x) x %in% c("state", "poverty", "female_house"))
cols_to_del <- cols |> append("", after = 0)
model_selection <- tibble(to_del = cols_to_del) |>
    mutate(
        model = str_c(
            "poverty ~ ",
            map(to_del,
                \(y) str_flatten(
                    purrr::discard(
                        cols,
                        \(x) x == y),
                    collapse = " + ")
                )
            ),
        adj_r_sq = map(
            model,
            \(x) round(summary(
                lm(formula(x), data = states)
            )$adj.r.squared, digits = 2))
    )
kable(model_selection)
```

to_del	model	adj_r_sq
	poverty ~ metro_res + white + hs_grad	0.62
metro_res	poverty ~ white + hs_grad	0.56
white	poverty ~ metro_res + hs_grad	0.58
hs_grad	poverty ~ metro_res + white	0.17

Из этой таблицы мы видим, что лучший результат дает модель, где отсутствует только female_house.

Посмотрим на результаты этой модели.

```
summary(lm(poverty ~ metro_res + white + hs_grad,
    data = states))
```

```
##
## Call:
## lm(formula = poverty ~ metro_res + white + hs_grad, data = states)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2346 -1.4785 -0.3699  1.2153  4.5579
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 68.72202    6.38893  10.756 2.89e-14 ***
## metro_res   -0.05553    0.01898  -2.926  0.00528 **
## white       -0.05333    0.02148  -2.483  0.01665 *
## hs_grad     -0.56972    0.07527  -7.569 1.13e-09 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.915 on 47 degrees of freedom
## Multiple R-squared:  0.6412, Adjusted R-squared:  0.6183
## F-statistic:    28 on 3 and 47 DF,  p-value: 1.553e-10
```

Обратим внимание, что исключив переменную female_house, переменная white так же стала статистически взаимосвязана.

Завершая анализ этой модели, убедимся, что с остатками так же все в порядке.

```
states_final_lm <- lm(
    poverty ~ metro_res + white + hs_grad,
    data = states)
states_fin_resid <- states |>
    add_residuals(states_final_lm) |>
    add_predictions(states_final_lm)
grid.arrange(
    ggplot(states_fin_resid, aes(pred, resid)) +
        geom_point() +
        geom_hline(yintercept = 0, linetype = 2, color = "red"),
    ggplot(states_fin_resid, aes(sample = resid)) +
        geom_qq() +
        geom_qq_line(color = "red"),
    nrow = 1, ncol = 2
)
```