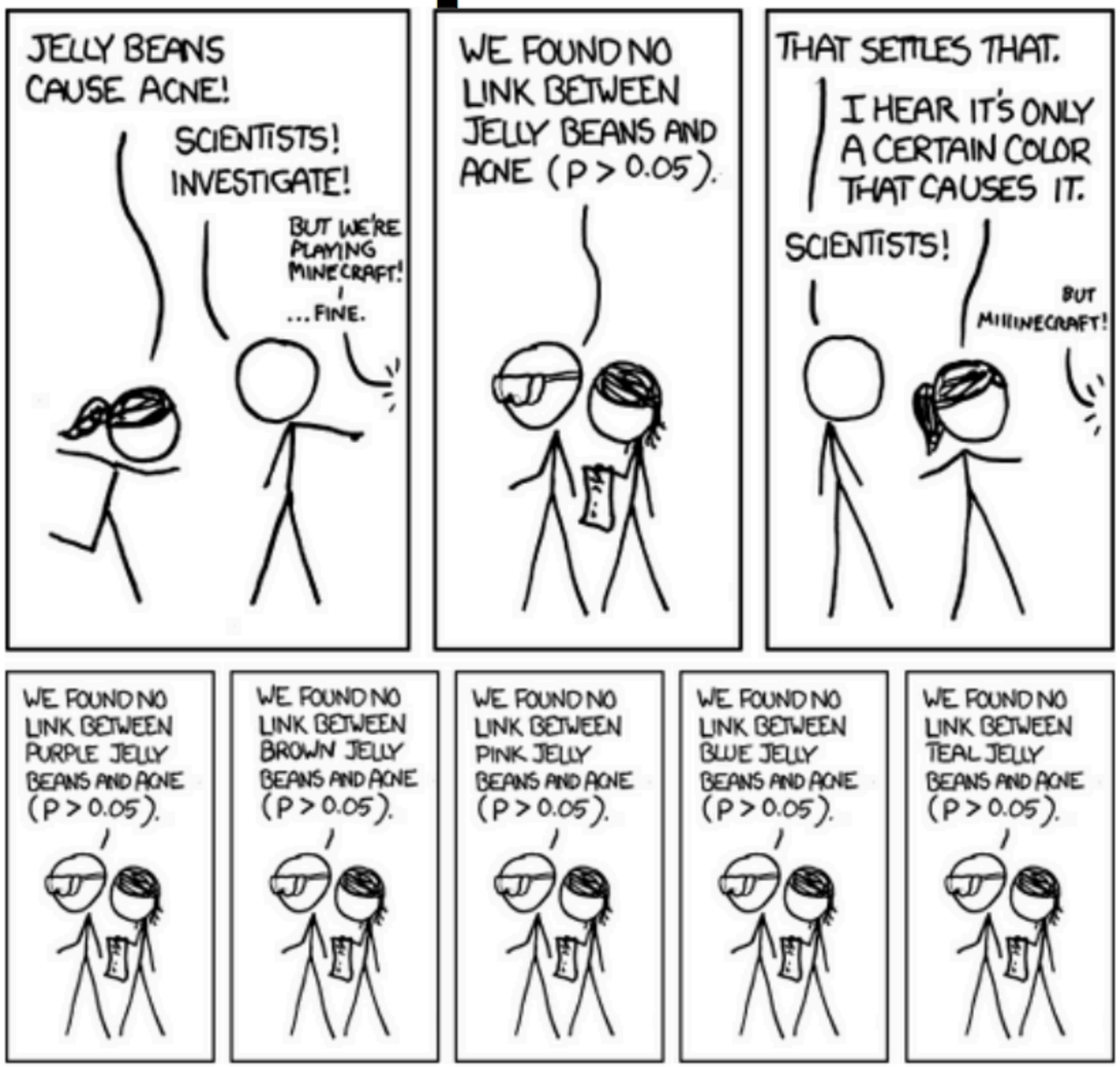
Множественное тестирование

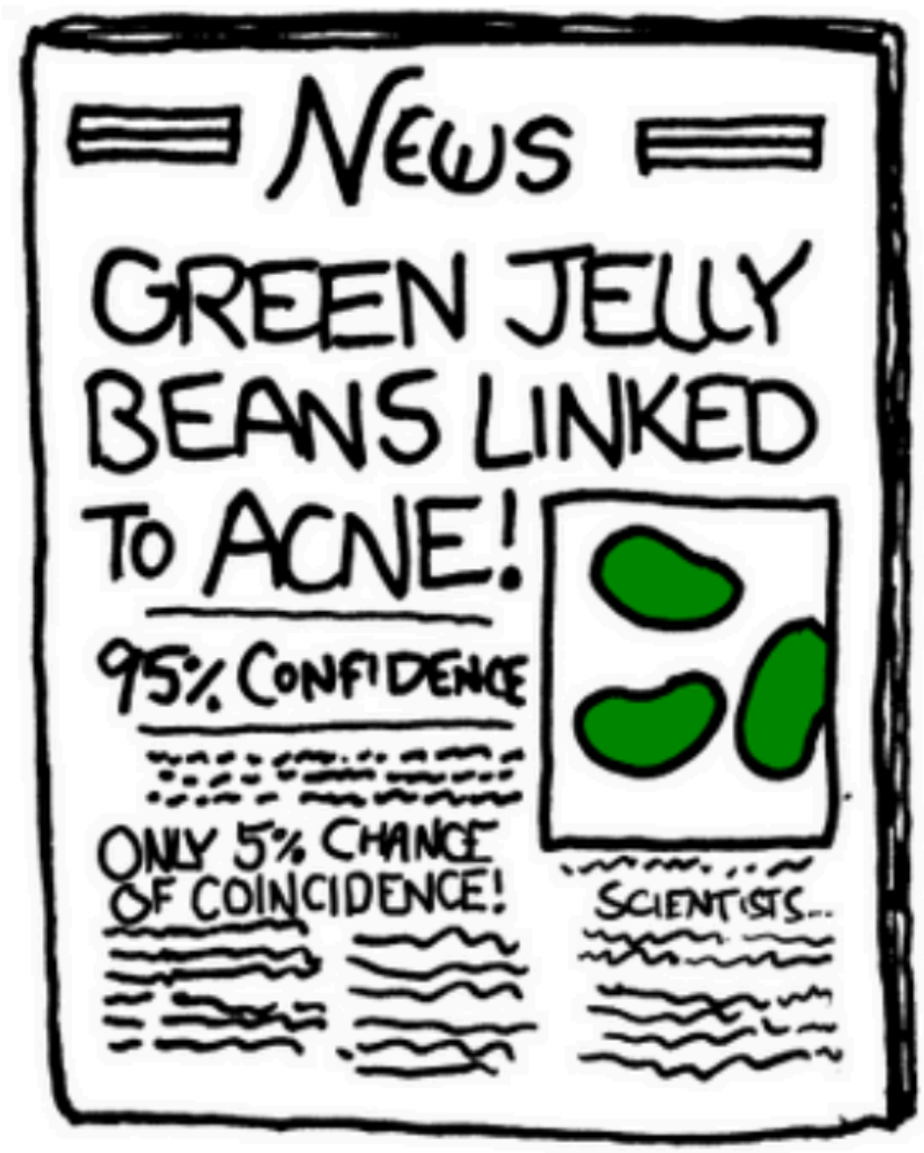
Не ищите того, чего нет. А то в конечном итоге - найдете. А это будет ошибка первого рода

@Игорь

Множественное тестирование



Множественное тестирование



Множественное тестирование

Рассмотрим датасет с 30000 генов, в котором нет ни одного дифференциально экспрессирующегося гена

Проведем t-test для каждого гена. Будем считать ген дифференциально экспрессируемым если $p < 0.05$.

Какова вероятность, что ни один ген не будет помечен как дифференциально экспрессируемый?

Сколько в среднем генов будет помечено как дифференциально экспрессируемые?

Поправки

- FWER (Family-Wise Error Rate) - вероятность, что среди отобранных генов хотя бы один ложно-положительный ген меньше заданного порога (0.05, например)

- FDR (False Discovery Rate) - процент ложно-положительных генов среди отобранных не больше, например, 20%

Смысл alpha разный для двух подходов

FWER

test	p-value
test1	p-value1
test2	p-value2
…	…
testN	p-valueN

Наша изначальная таблица

test	k	p-value
test1’	1	p-value1’
test2’	2	p-value2’
…	…	…
testM’	M	p-valueN’

Тесты, для которых мы отвергаем H0.

Гарантируем, что вероятность того, что во всей отобранной таблице встретится хотя бы один тест, для которого мы ошибочно отвергли H0 - alpha

FWER

One-step procedures:

1) Sidak correction
2) Bonferonni correction

Step-down procedures:

1) Holm-Sidak correction
2) Holm-Bonferonni correction

Step-up procedures:

Hochberg correction **Не рассматриваем**

FDR

test	p-value
test1	p-value1
test2	p-value2
…	…
testN	p-valueN

test	k	p-value
test1’	1	p-value1’
test2’	2	p-value2’
…	…	…
testM’	M	p-valueN’

Наша изначальная таблица

Тесты, для которых мы отвергаем H0. Гарантируем, что доля генов, для которых мы ошибочно отвергли H0 - alpha

p.adjust {stats}

Adjust P-values for Multiple Comparisons

Description

Given a set of p-values, returns p-values adjusted using one of several methods.

Usage

```
p.adjust(p, method = p.adjust.methods, n = length(p))

p.adjust.methods
# c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY",
#   "fdr", "none")
```

Возвращает скорректированные (adjusted) p-value
Это p-value, при сравнении которых с вашим alpha, меньше alpha окажутся те p-value, которые были бы отобраны соответствующим методом

Adjusted p-value

p-value, при сравнении которых с вашим alpha, меньше alpha окажутся те p-value, которые были бы отобраны соответствующим методом

Пример: сколько p-value из списка 0.01, 0.05, 0.04, 0.03, 0.001, 0.015, 0.20 останутся значимыми после поправки Холма-Бонферонни на уровне значимости alpha=0.05

```
alpha <- 0.05
pvals <-c(0.01, 0.05, 0.04, 0.03, 0.001, 0.015, 0.20)
adj_pvals <- p.adjust(pvals, method = 'holm')
sum(adj_pvals < 0.05)
```

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

Adjusted p-value
One-step procedure

$$p < \frac{alpha}{N} = thres$$

На примере Бонферонни, мы можем записать следующее условие иначе

$$adjust\_p = p \cdot N < \alpha$$

Adjusted p-value
Step-down procedure
Неверный подход

$$p_k < \frac{\alpha}{N-k+1} = thres(k)$$

На примере Холма Бонферонни, мы можем записать следующее условие иначе

$$adjust\_p_k = p_k \cdot (N-k+1) < \alpha$$

Adjusted p-value

Step-down procedure

Верный подход

$$p_k < \frac{\alpha}{N-k+1} = thres(k)$$

На примере Холма Бонферонни, мы можем записать следующее условие иначе

$$adjust\_p_1 = p_1 \cdot N < \alpha$$

$$adjust\_p_k = max(p_k \cdot (N-k+1),\ adjust\_p_{k-1}\ )) < \alpha$$

Adjusted p-value

Step-down procedure.

Пример, Холм-Бонферонни

test	p-value	Порядок в сортировке (k)	adjusted p-value
test1	0.01	2	
test2	0.05	6	
test3	0.04	5	
test4	0.03	4	
test5	0.001	1	
test6	0.015	3	
test7	0.20	7	

Adjusted p-value

Step-down procedure.

Пример, Холм-Бонферонни

test	p-value	Порядок в сортировке (k)	adjusted p-value
test1	0.01	2	
test2	0.05	6	
test3	0.04	5	
test4	0.03	4	
test5	0.001	1	0.001 * 7 = 0.007
test6	0.015	3	
test7	0.20	7	

Adjusted p-value

Step-down procedure.

Пример, Холм-Бонферонни

test	p-value	Порядок в сортировке (k)	adjusted p-value
test1	0.01	2	max (0.007, 0.01 * 6)=0.06
test2	0.05	6	
test3	0.04	5	
test4	0.03	4	
test5	0.001	1	0.001 * 7 = 0.007
test6	0.015	3	
test7	0.20	7	

Adjusted p-value

Step-down procedure.

Пример, Холм-Бонферонни

test	p-value	Порядок в сортировке (k)	adjusted p-value
test1	0.01	2	0.06
test2	0.05	6	
test3	0.04	5	
test4	0.03	4	
test5	0.001	1	0.007
test6	0.015	3	max (0.06, 0.015 * 5)=0.075
test7	0.20	7	

Adjusted p-value

Step-down procedure.

Пример, Холм-Бонферонни

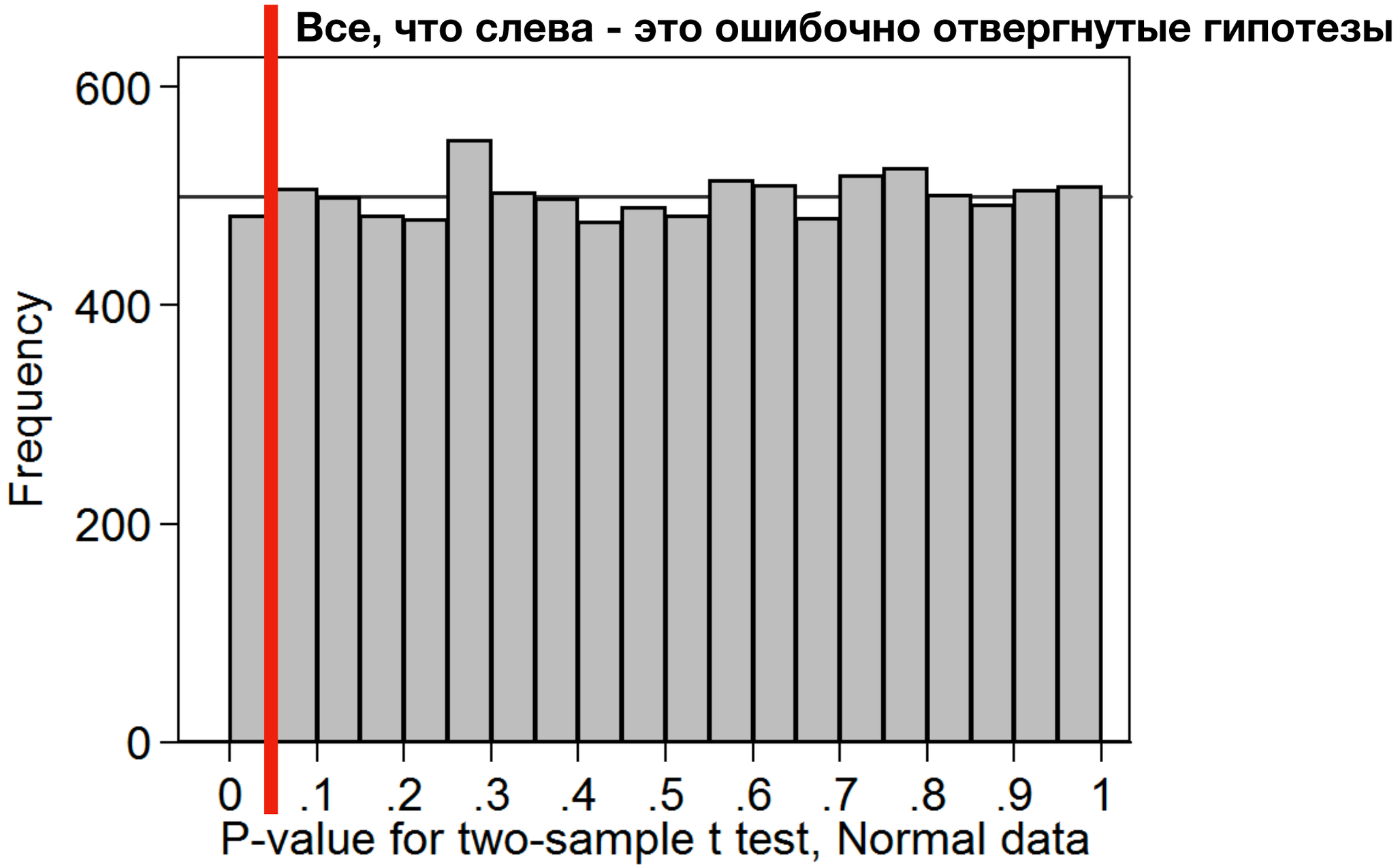
test	p-value	Порядок в сортировке (k)	adjusted p-value
test1	0.01	2	0.06
test2	0.05	6	0.120
test3	0.04	5	0.120
test4	0.03	4	0.120
test5	0.001	1	0.007
test6	0.015	3	0.075
test7	0.20	7	0.200

Где можно встретить проблему множественного тестирования?

- Проводится много тестов
- Считается много корреляций
- Строится много разных моделей
- Проводится много экспериментов
- Везде. Потому более корректное название - multiplicity problem. Всегда, когда вы делаете что-то больше одного раза, стоит проверить, а не возникла ли проблема множественного тестирования

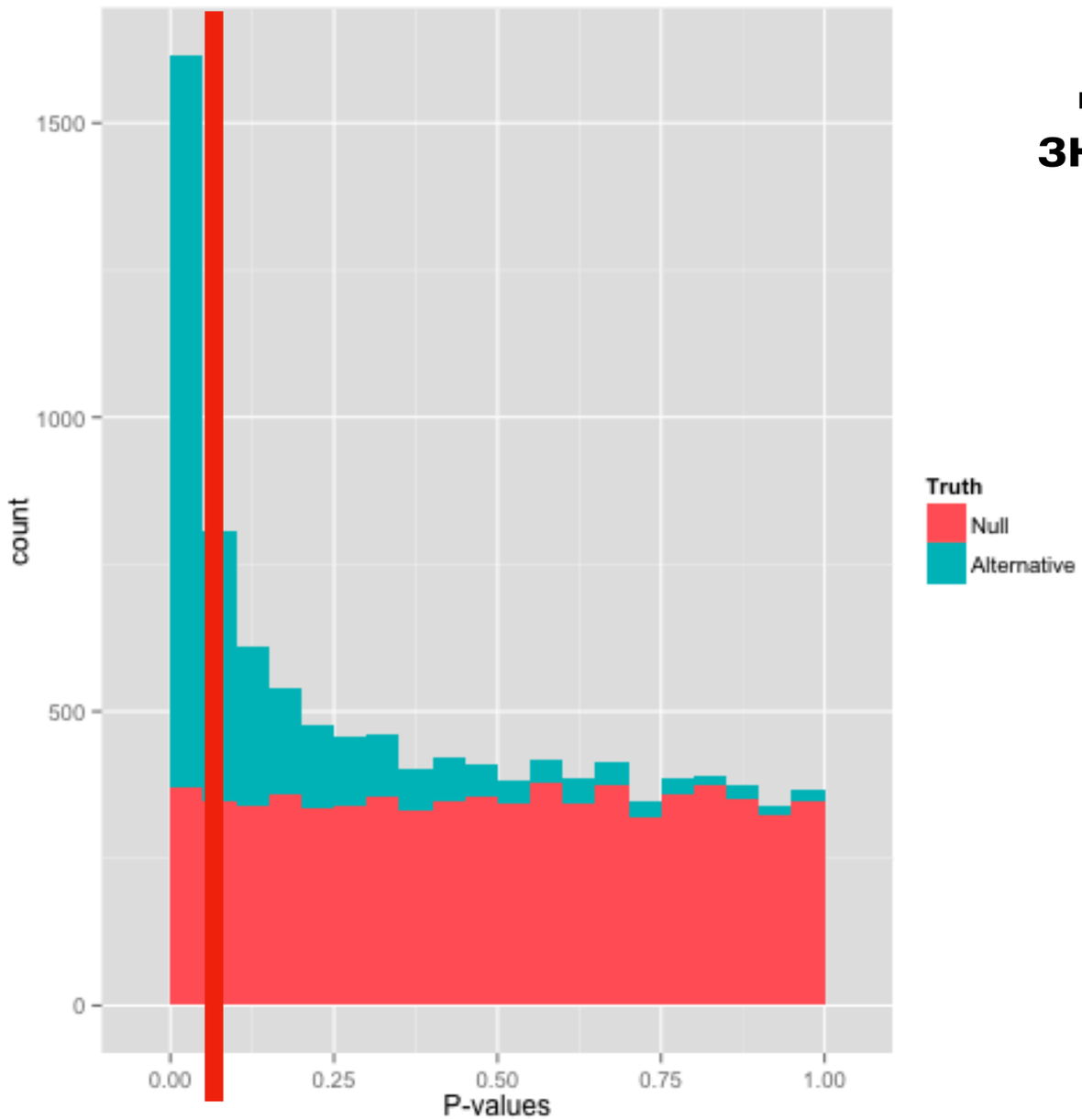
Где можно встретить проблему множественного тестирования?

- Проводится много тестов



Где можно встретить проблему множественного тестирования?

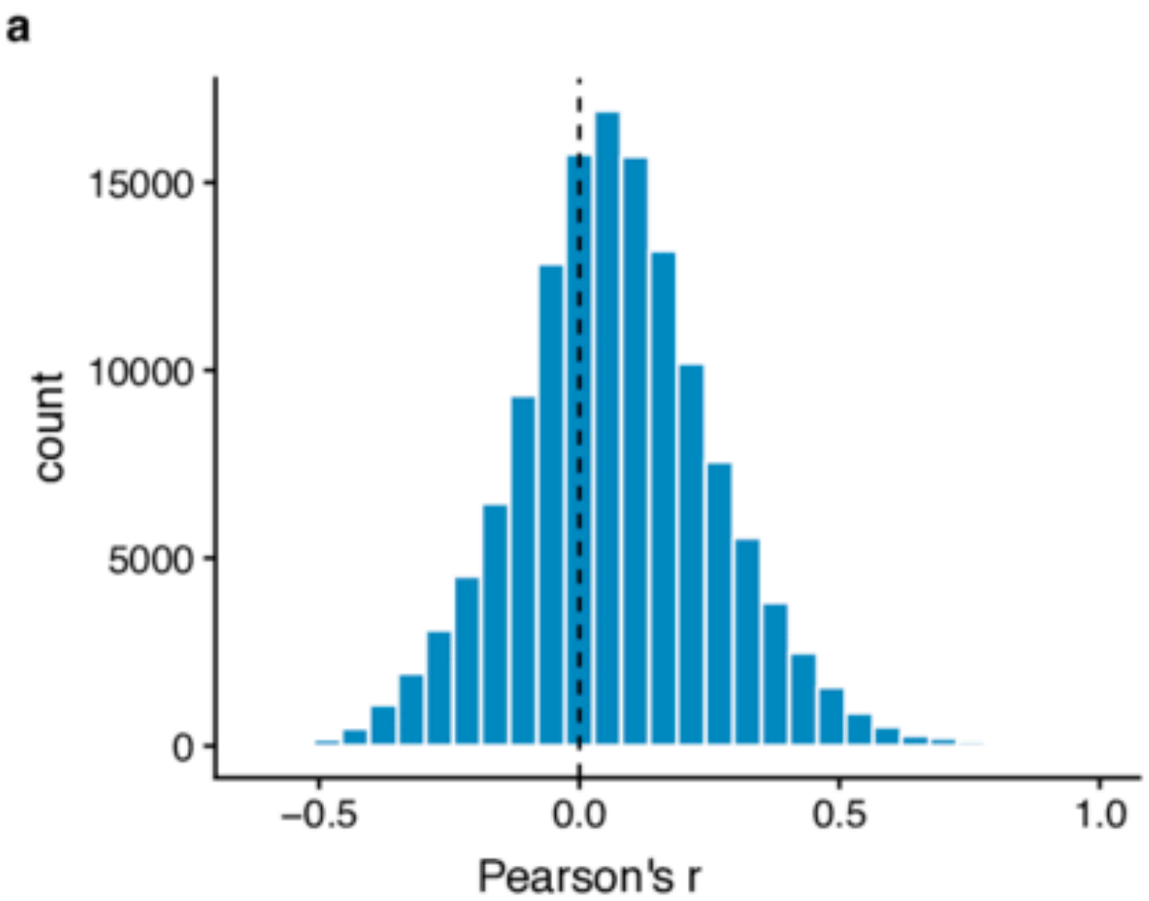
- Проводится много тестов



Даже если есть реально значимые результаты - они могут быть разбавлены незначимыми

Где можно встретить проблему множественного тестирования?

- Считается много корреляций



Где можно встретить проблему множественного тестирования?

- Проводится много тестов
- Считается много корреляций
- Строится много разных моделей
- Проводится много экспериментов
- Везде. Потому более корректное название - multiplicity problem. Всегда, когда вы делаете что-то больше одного раза, стоит проверить, а не возникла ли проблема множественного тестирования

Как бороться?

- Поправки на множественное тестирование. Не всегда применимы - как делать поправку на то, что протестировал 100 разных моделей машинного обучения?

Как бороться?

- Процедура, подвергающаяся риску множественного тестирования - лишь часть пайплайна

Сырые данные

первичные (нестрогие) фильтры

Данные

Строгие фильтры

Результат

Как бороться?

- Проверка на новых данных. Как вариант - отложить часть данных и не использовать их в пайплайне. Проверить на них полученные выводы

Train-test split

Обучение (train) **Тест (test)**

Проводим наш анализ на train, проверяем на test. Exploratory data analysis

Train-test split

Обучение (train) **Тест (test)**

Обучаем модель на train, проверяем качество модели на test.

Гиперпараметры

- У модели есть параметры и гиперпараметры
- Параметры модели учатся на основе выборки самой моделью (алгоритмом ее обучения)
- Гиперпараметры - это параметры, которые задаем мы и которые влияют на то, как модель учит параметры

Примеры гиперпараметров

1. **Тип модели**
2. **Признаки, которые в ней используем**
3. **Аргументы, влияющие на то, как именно подбираются параметры/веса модели**

Train-test split?

Обучение (train) **Тест (test)**

Обучаем модель на train, проверяем качество модели на test.

Как подбирать гиперпараметры модели? - Никак

Train-validation-test split!

Обучение (train) **Валидация (validation)** **Тест (test)**

1) **Выбираем некоторые значения гиперпараметров**
2) **Обучаем модель с такими гиперпараметрами на train**
3) **Смотрим качество на validation**
4) **Пробуем таким образом много разных значений гиперпараметров и выбираем то, которое дает наилучшее**

Train-validation-test split!

Обучение (train) **Валидация (validation)** **Тест (test)**

1) **Выбираем некоторые значения гиперпараметров**
2) **Обучаем модель с такими гиперпараметрами на train**
3) **Смотрим качество на validation**
4) **Пробуем таким образом много разных значений гиперпараметров и выбираем то, которое дает наилучшее**

Какие минусы подхода?

Train-validation-test split?

Обучение (train) **Валидация (validation)** **Тест (test)**

Какие минусы подхода?

1) **Существенно уменьшаем объем данных, на которых учится модель**
2) **Большая нестабильность оценки качества при сравнении моделей из-за малого размера выборки**

Кросс-валидация

Обучение (train)

Тест (test)

...

Много разбиений на train и вариацию. На каждом разбиении выбираем лучшие гиперпараметры. Потом смотрим, какие значения гиперпараметров встречаются чаще всего, на основании чего делаем вывод об итоговых значениях гиперпараметров

Что еще можно оценить?

Кросс-валидация

Обучение (train)

Тест (test)

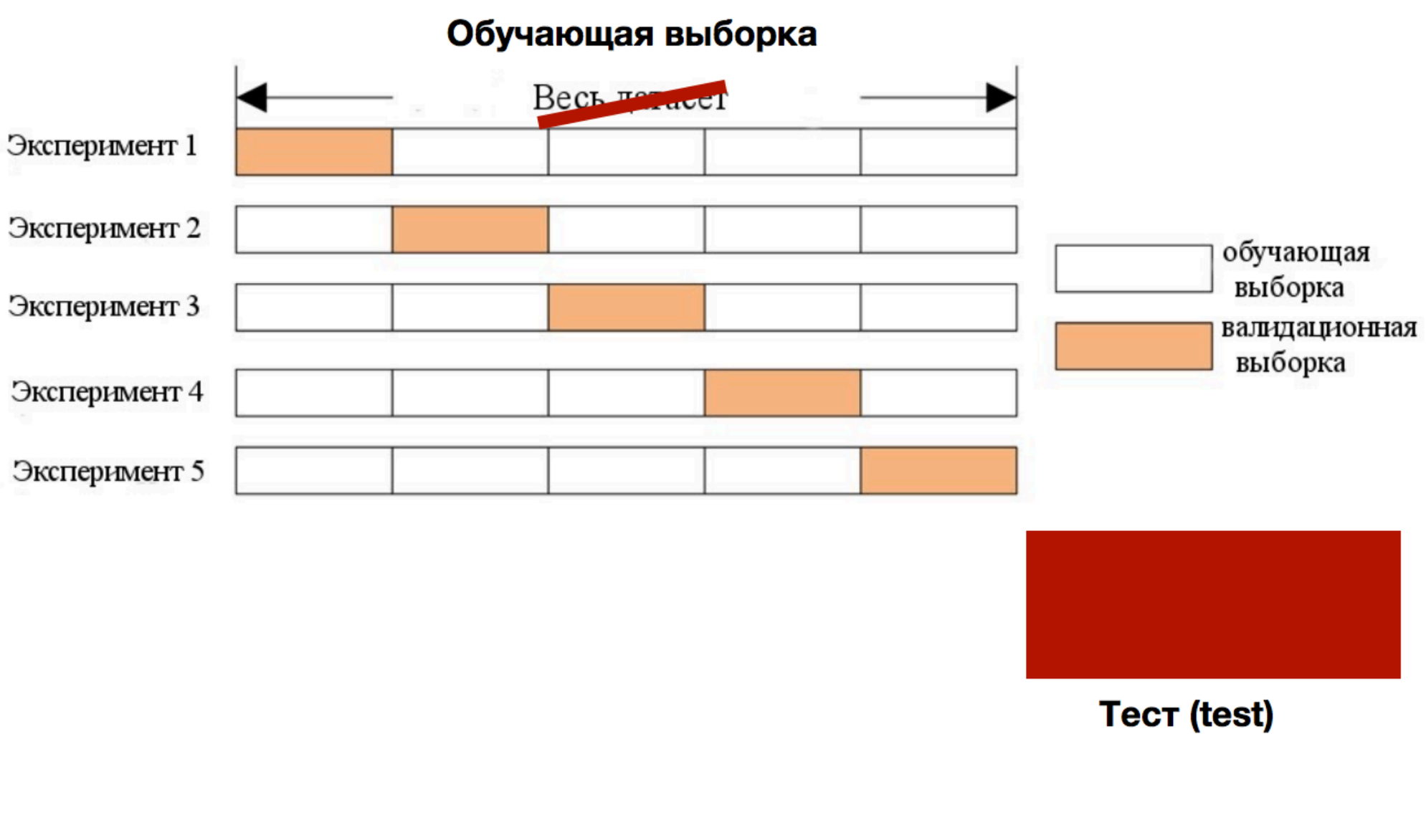
...

Что еще можно оценить?

Для данного набора значений гиперпараметров можем оценить среднее качество модели и дисперсию по разным разбиениям

Кросс-валидация. K-fold кросс-валидация

Тест отдельно должен быть



Парадокс Симпсона

	Количество кандидатов	**Доля поступивших**
Мужчины	8442	**44%**
Женщины	4321	35%

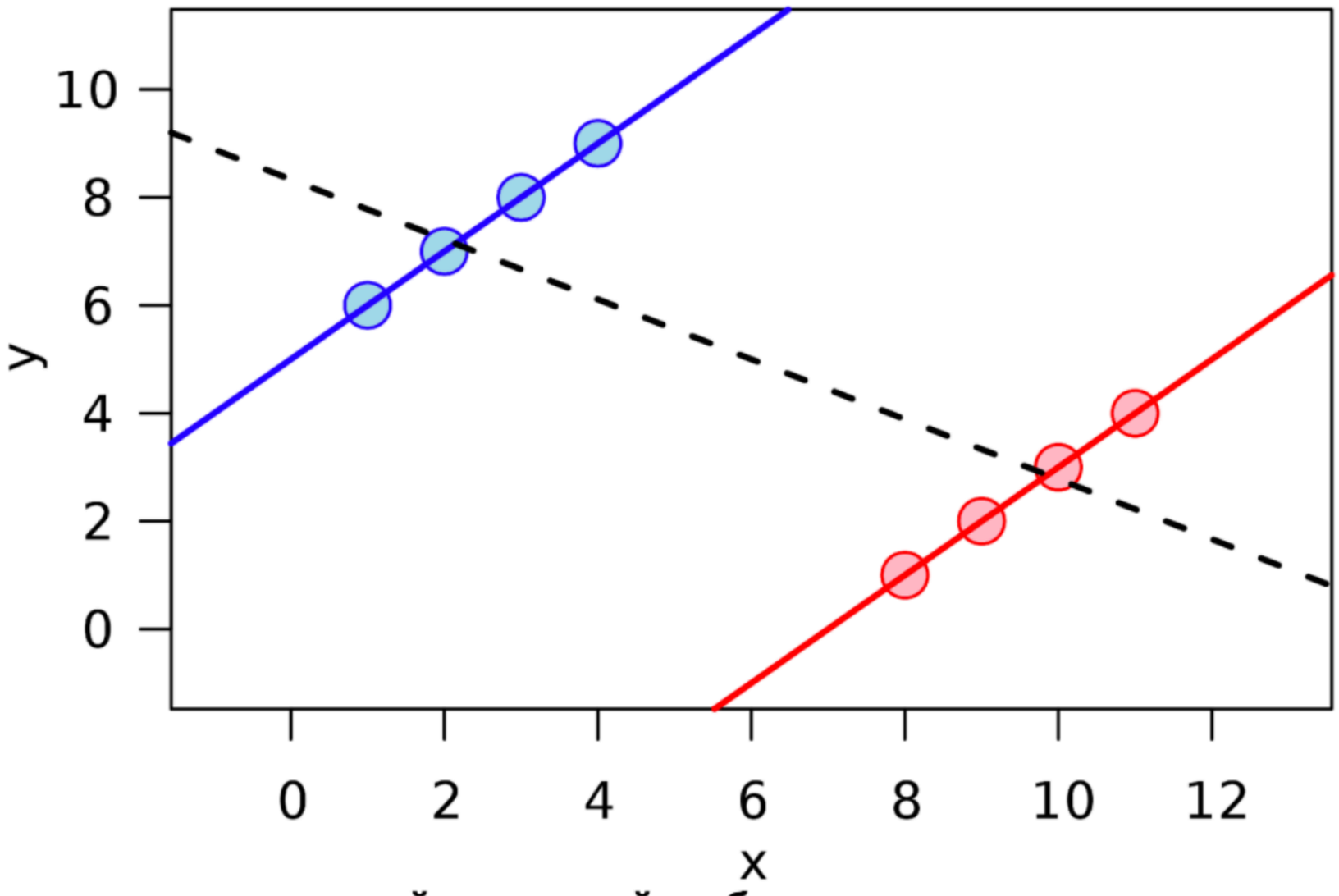
Дискриминируют женщин

Парадокс Симпсона

	Мужчины		**Женщины**	
	Количество кандидатов	**Доля поступивших**	Количество кандидатов	**Доля поступивших**
Кафедра 1	825	62%	108	**82%**
Кафедра 2	560	63%	25	**68%**
Кафедра 3	325	**37%**	593	34%
Кафедра 4	417	33%	375	**35%**
Кафедра 5	191	**28%**	393	24%

Дискриминируют мужчин)

Парадокс Симпсона



Для разделенных синей и красной выборок корреляция имеет один знак, для объединенных - разный

Категориальные переменные в линейных моделях

Предсказываем цены на ноутбуки

```
laptop <- read.csv("laptop_price.csv")
head(laptop)
```

```
##   Manufacturer  Model Processor Memory_Gb HDD_Gb HDD_type Price_RUR
## 1        Acer Aspire  i3-3110M         4    500      HDD     16400
## 2        Acer Aspire  i3-3120M         4    500      HDD     16500
## 3        Acer Aspire  i5-3230M         4    500      HDD     18500
## 4        Acer Aspire   \xd1-70         2    500      HDD     12000
## 5        Acer Aspire   \xd1-70         2    500      HDD     12000
## 6        Acer Aspire     1007U         2    500      HDD     11300
##   Screen_size_inch Battery_capacity_mAh   OS     Color
## 1             15.6                4400 Win8     black
## 2             15.6                4400 Win8     black
## 3             15.6                4400 Win8     black
## 4             11.6                2500 Win8 turquoise
## 5             11.6                2500 Win8     black
## 6             11.6                5000 Win8 turquoise
```

Просто память

```
model <- lm(Price_RUR ~ Memory_Gb, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ Memory_Gb, data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -16662  -6292  -2558    790  64438
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5022.8     1651.2   3.042  0.00256 **
## Memory_Gb     4442.4      333.1  13.335  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12310 on 304 degrees of freedom
## Multiple R-squared:  0.3691, Adjusted R-squared:  0.367
## F-statistic: 177.8 on 1 and 304 DF,  p-value: < 2.2e-16
```

Несколько переменных

```
model <- lm(Price_RUR ~ Memory_Gb +  Screen_size_inch + HDD_Gb, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ Memory_Gb + Screen_size_inch + HDD_Gb,
##     data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -24879  -5552  -1505   2670  61021
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      16475.481   4437.238   3.713 0.000244 ***
## Memory_Gb         7266.167    406.667  17.868  < 2e-16 ***
## Screen_size_inch  -511.390    350.186  -1.460 0.145237
## HDD_Gb             -31.022      3.305  -9.387  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10650 on 302 degrees of freedom
## Multiple R-squared:  0.5308, Adjusted R-squared:  0.5262
## F-statistic: 113.9 on 3 and 302 DF,  p-value: < 2.2e-16
```

Факторная переменная

```
model <- lm(Price_RUR ~ Manufacturer, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ Manufacturer, data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32377  -8255  -2499   3490  73174
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        21198.6     1415.0  14.981  < 2e-16 ***
## ManufacturerApple  46078.5     3527.5  13.063  < 2e-16 ***
## ManufacturerAsus     206.2     1714.4   0.120 0.904357
## ManufacturerDell    7427.6     2079.1   3.573 0.000411 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12090 on 302 degrees of freedom
## Multiple R-squared:  0.3958, Adjusted R-squared:  0.3898
## F-statistic: 65.95 on 3 and 302 DF,  p-value: < 2.2e-16
```

Кодирование меток

A/G -> 0, T/C -> 1, ...

Какой минус?

Кодирование меток

A/G -> 0, T/C -> 1, ...

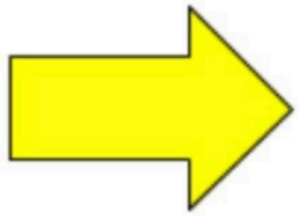
Какой минус?

Задаем неявную информацию о том, что A/G > T/C и тд

Использовать только вместе с сортировкой по предсказываемой величине

One-hot encoding

Цвет
Красный
Красный
Желтый
Зеленый
Желтый



Красный	Желтый	Зеленый
1	0	0
1	0	0
0	1	0
0	0	1

Факторная переменная

- Влияет ли цвет ноутбука на его цену?
- Модель, если x – число: $y_i = \alpha x_{1i} + \beta x_{2i} + \varepsilon_i$
- Если x – фактор, то такая запись не подходит. Вместо этого:

$$y_i = \alpha_1 I(x_{1i}==black) + \alpha_2 I(x_{1i}==white) + \ldots + \varepsilon_i$$

Коэффициент (подбираются при построении модели)

Индикатор (равен 1, если x – черный цвет, иначе 0)

Если две факторные переменные?

$$y_i = \alpha_1 I(x_{1i}==black) + \alpha_2 I(x_{1i}==white) + \ldots +$$

$$+ \beta_1 I(x_{2i}==Apple) + \beta_2 I(x_{2i}==ASUS) + \ldots + \varepsilon_i$$

Факторная переменная

```
model <- lm(Price_RUR ~ Manufacturer, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ Manufacturer, data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32377  -8255  -2499   3490  73174
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        21198.6     1415.0  14.981  < 2e-16 ***
## ManufacturerApple  46078.5     3527.5  13.063  < 2e-16 ***
## ManufacturerAsus     206.2     1714.4   0.120 0.904357
## ManufacturerDell    7427.6     2079.1   3.573 0.000411 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12090 on 302 degrees of freedom
## Multiple R-squared:  0.3958, Adjusted R-squared:  0.3898
## F-statistic: 65.95 on 3 and 302 DF,  p-value: < 2.2e-16
```

Почему на одно меньше, чем производителей?

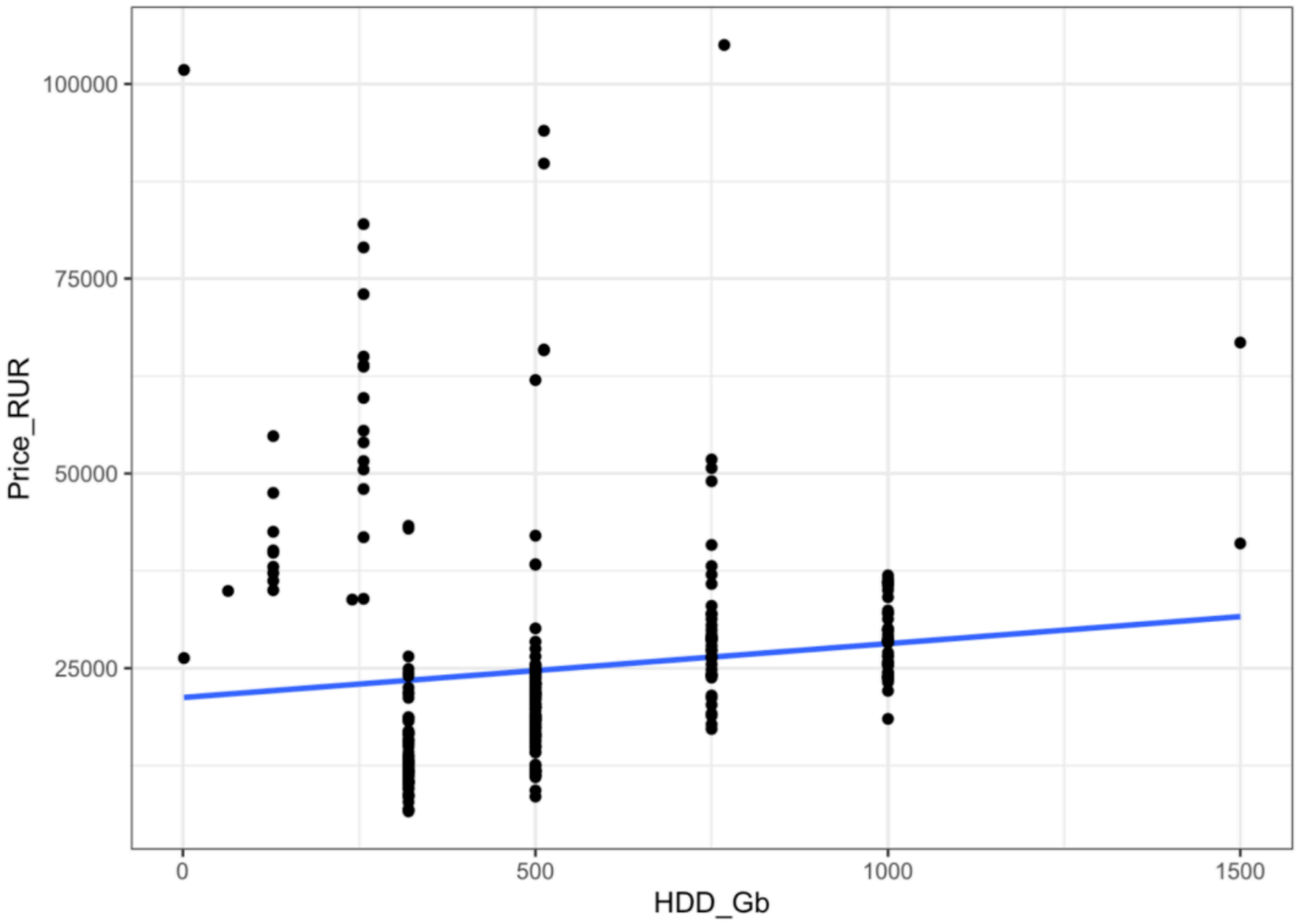
Числа и факторы

```
model <- lm(Price_RUR ~ HDD_Gb, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ HDD_Gb, data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -16851  -9244  -3445   1912  80551
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21238.584   2027.553  10.475   <2e-16 ***
## HDD_Gb          6.913      3.410   2.027   0.0435 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15400 on 304 degrees of freedom
## Multiple R-squared:  0.01334,    Adjusted R-squared:  0.01009
## F-statistic: 4.109 on 1 and 304 DF,  p-value: 0.04352
```

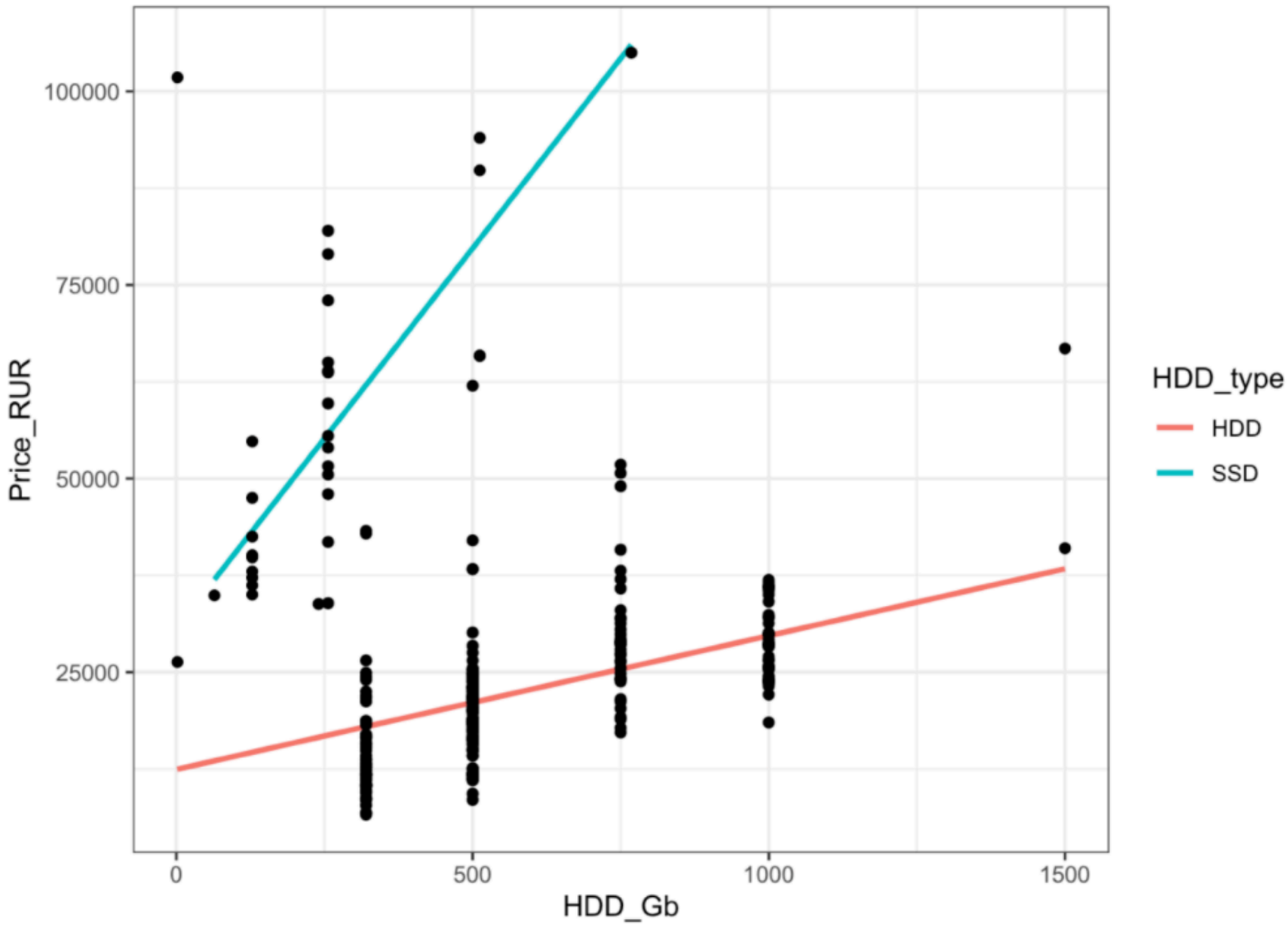
```
laptop %>% ggplot(aes(y=Price_RUR, x=HDD_Gb)) + geom_smooth(method="lm", se=F) + geom_point() + theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
```



```
laptop %>% ggplot(aes(y=Price_RUR, x=HDD_Gb)) + geom_smooth(aes(color=HDD_type), method="lm", se=F) + geom_point
() + theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Числа и факторы

```
model <- lm(Price_RUR ~ HDD_Gb + HDD_type, data=laptop)
summary(model)
```

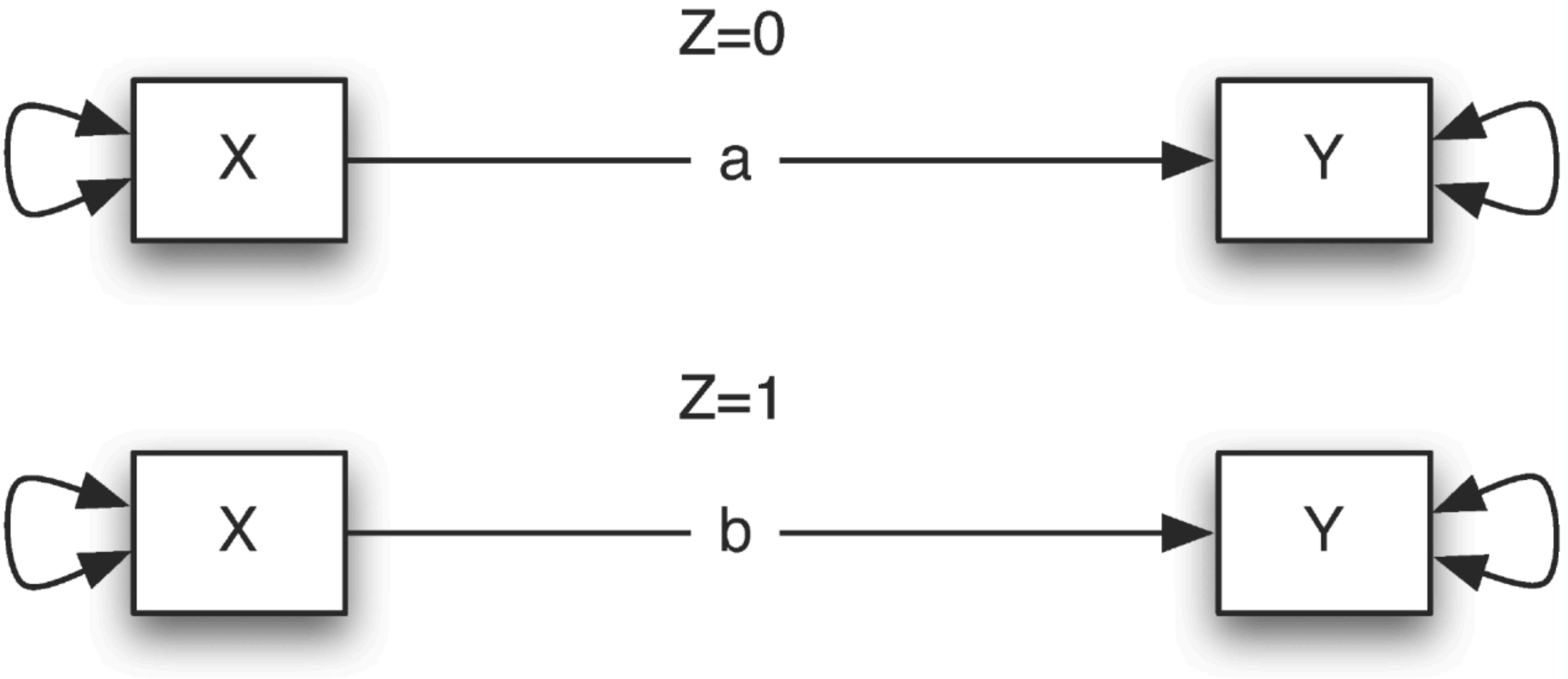
```
##
## Call:
## lm(formula = Price_RUR ~ HDD_Gb + HDD_type, data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -22833  -5948  -1886   2889  91028
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10741.160   1594.347   6.737 8.14e-11 ***
## HDD_Gb         20.290      2.591   7.830 8.27e-14 ***
## HDD_typeSSD 40797.575   2442.199  16.705  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11130 on 303 degrees of freedom
## Multiple R-squared:  0.4864, Adjusted R-squared:  0.483
## F-statistic: 143.5 on 2 and 303 DF,  p-value: < 2.2e-16
```

Числа и факторы

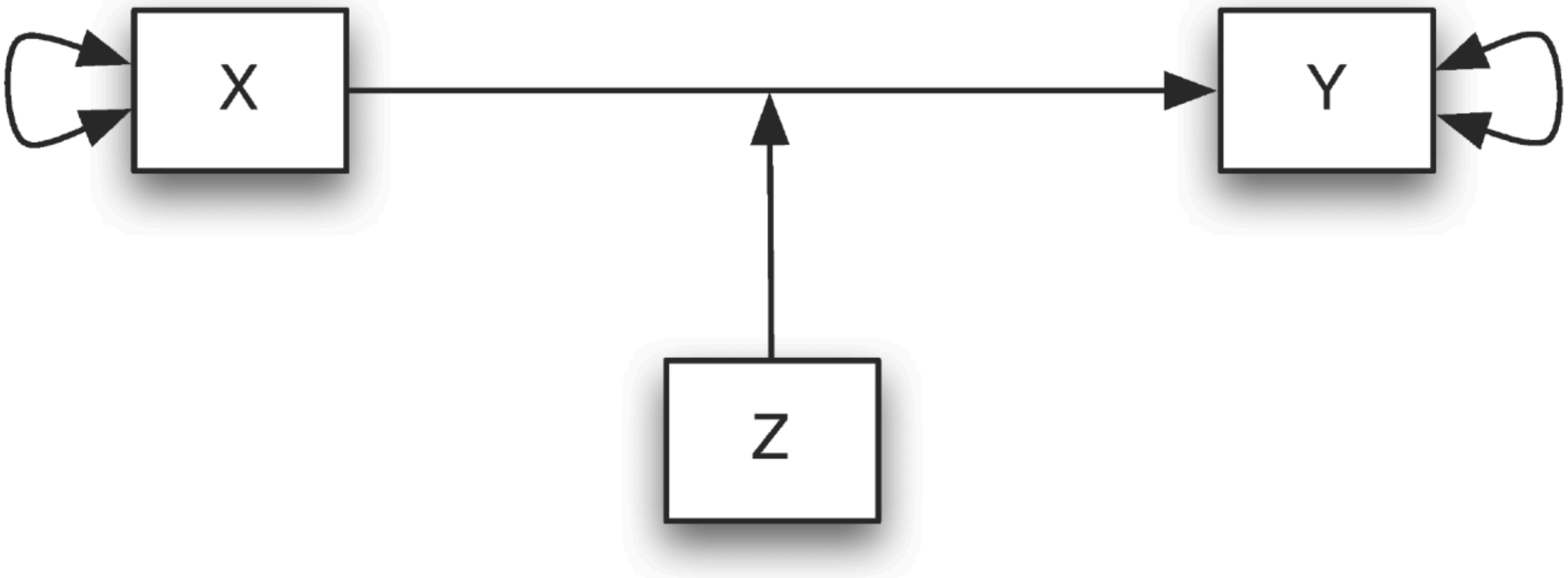
```
model <- lm(Price_RUR ~ HDD_Gb + HDD_type + HDD_Gb:HDD_type, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ HDD_Gb + HDD_type + HDD_Gb:HDD_type,
##     data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -21886  -6049  -1461   2885  89344
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       12430.529   1525.776   8.147 9.97e-15 ***
## HDD_Gb               17.270      2.488   6.941 2.38e-11 ***
## HDD_typeSSD       18232.081   4265.934   4.274 2.58e-05 ***
## HDD_Gb:HDD_typeSSD    80.870     12.874   6.281 1.17e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10480 on 302 degrees of freedom
## Multiple R-squared:  0.5457, Adjusted R-squared:  0.5412
## F-statistic: 120.9 on 3 and 302 DF,  p-value: < 2.2e-16
```

Moderation effect



Moderation effect



Как выглядит линейная модель

$$Y = \beta_0 + \beta_1 \cdot X + \beta_2 \cdot Z + \beta_3 \cdot X \cdot Z + \epsilon$$

Вклад X **Вклад Z** **moderation effect Z**

Как выглядит линейная модель

$$Y = \beta_0 + \beta_1 \cdot X + \beta_2 \cdot Z + \beta_3 \cdot X \cdot Z + \epsilon$$

Вклад X **Вклад Z** **moderation effect Z**

Если Z либо 0, либо 1, то как выглядит?

$$Y = \beta_0 + \beta_1 \cdot X + \epsilon, \quad Z = 0$$

$$Y = \beta_0 + \beta_2 + (\beta_1 + \beta_3) \cdot X + \epsilon, \quad Z = 1$$

Числа и факторы

```
model <- lm(Price_RUR ~ HDD_Gb * HDD_type, data=laptop)
summary(model)
```

```
##
## Call:
## lm(formula = Price_RUR ~ HDD_Gb * HDD_type, data = laptop)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -21886  -6049  -1461   2885  89344
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         12430.529   1525.776   8.147 9.97e-15 ***
## HDD_Gb                 17.270      2.488   6.941 2.38e-11 ***
## HDD_typeSSD         18232.081   4265.934   4.274 2.58e-05 ***
## HDD_Gb:HDD_typeSSD     80.870     12.874   6.281 1.17e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10480 on 302 degrees of freedom
## Multiple R-squared:  0.5457, Adjusted R-squared:  0.5412
## F-statistic: 120.9 on 3 and 302 DF,  p-value: < 2.2e-16
```

```r
library(psych)
```

```r
example <- lm(bdi ~ stateanx*epiNeur, data=epi.bfi)
example
```

```
##
## Call:
## lm(formula = bdi ~ stateanx * epiNeur, data = epi.bfi)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.0493  -2.2513  -0.4707   2.1135  11.9949
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.06367    2.18559   0.029   0.9768
## stateanx          0.03750    0.06062   0.619   0.5368
## epiNeur          -0.14765    0.18869  -0.782   0.4347
## stateanx:epiNeur  0.01528    0.00466   3.279   0.0012 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.12 on 227 degrees of freedom
## Multiple R-squared:  0.4978, Adjusted R-squared:  0.4912
## F-statistic: 75.02 on 3 and 227 DF,  p-value: < 2.2e-16
```