

# Вытекающие последствия

Разбор статистики еще раз!

## «Сложные» задачи: 6-8

(применение стат. тестов + merge + apply)

- Мало кто попробовал решать (а зря!)
- Те, кто решал, в основном не ошибались
- Техническая трудность (судя по вопросам на КР): проблема с чтением «нетипичных» файлов (например, с комментариями) -> **на текстовый файл полезно посмотреть глазами прежде чем загружать в R**
- Типичная ошибка: неправильное решение о принятии/отвержении Но при правильно посчитанном pvalue
- Путаница с парными и непарными тестами

# Парный или непарный тест?

- Сравниваются 2 **независимые** выборки (группа 1: здоровые, группа два: больные) – **тест непарный!**
- Сравниваются 2 **зависимые** выборки (группа 1: пациенты во время болезни, группа два: **те же** пациенты после выздоровления) – **тест парный!**

```
delays<-read.csv("delays_jun_dec.csv")
```

```
head(delays)
```

```
  X airport OnTimeJune DelayedJune OnTimeDec DelayedDec
1 1  ABE      203         43      167         44
2 2  ABI      183         17      178         22
3 3  ABQ     2307        504     1807        523
4 4  ABR       59          1       49         12
5 5  ABY       68         15       68         16
6 6  ACT      174         42      102          9
```

•

•

```
>row.fisher<-function(x){  
+m<-matrix(x, ncol=2)  
+ft<-fisher.test(m)  
+ft$p.value}
```

- Применяем к нашей таблице:

```
>delay.fisher<-apply(delays[,3:ncol(delays)], 1, row.fisher)  
> head(delay.fisher)
```

- [1] 4.032596e-01 5.006086e-01 6.407614e-05 2.054603e-03  
1.000000e+00
- [6] 9.410315e-03

Или

- ```
pValues <- rep(NA,nrow(delays))  
for(i in 1:nrow(delays)){ pValues[i]<-  
row.fisher(delays[i, 3:ncol(delays)]);}
```

```
> which(delay.fisher<3.5e-05)
```

```
[1] 17 18 46 62 79 86 96 135 136 145 150  
154 156 172 173 190
```

```
[17] 196 200 206 207 212 219 223 230 231 232  
239 241 253 255 256 257
```

```
[33] 274 281
```

```
> length(which(delay.fisher<3.5e-05))
```

```
[1] 34
```