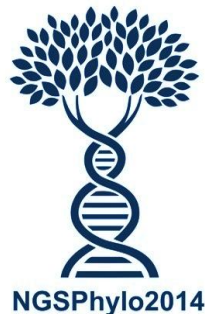


Язык R

ЛЕКЦИЯ 2

Артем Артемов, Светлана Виноградова, Елена Ставровская
3 октября 2014



ИЛС
ИнтерЛабСервис





матрицы

списки

СТРУКТУРЫ ДАННЫХ (КОНТЕЙНЕРЫ)

Матрицы

- На вид трудно отличить от data frame
- «Двумерный вектор»
- Все элементы одного типа

Матрицы заполняются по колонкам:

```
> m <- matrix(c(1:6), nrow=2, ncol=3)
```

```
> m
```

	[,1]	[,2]	[,3]
[1,]	1	3	5
[2,]	2	4	6

Или по рядам (аргумент **byrow=T**)

```
> m <- matrix(c(1:6), nrow=2, ncol=3, byrow=T)
```

```
> m
```

	[,1]	[,2]	[,3]
[1,]	1	2	3
[2,]	4	5	6

```
> as.matrix(1:3)
```

	[,1]
[1,]	1
[2,]	2
[3,]	3

Матрицы

- Обращение к элементам – как в data frame

```
> m[2,3]
```

```
[1] 6
```

```
> m[2,]
```

```
[1] 2 4 6
```

```
> m[,3]
```

```
[1] 5 6
```

```
> m[2,3] = 1
```

```
> m
```

	[,1]	[,2]	[,3]
[1,]	1	3	5
[2,]	2	4	6

Матрицы

```
> m <- matrix(nrow=2, ncol=3)
```

```
> m
```

	[,1]	[,2]	[,3]
[1,]	NA	NA	NA
[2,]	NA	NA	NA

```
> dim(m)
```

← Атрибут **dim** – размер матрицы
См. также **nrow**, **ncol**

```
[1] 2 3
```

```
> attributes(m)
```

```
$dim
```

```
[1] 2 3
```

Матрицы

Матрица – это вектор с атрибутом *dim*

```
> v <- 1:6
```

```
> dim(v)
```

количество строк

NULL

```
> dim(v) <- c(2,3)
```

количество столбцов

```
> v
```

	[,1]	[,2]	[,3]
[1,]	1	3	5
[2,]	2	4	6

```
> is.matrix(v)
```

```
[1] TRUE
```

Матрицы

```
> a <- c(1:3)
```

```
> b <- c(11:13)
```

```
> cbind(a, b) # “column bind”
```

	a	b
[1,]	1	11
[2,]	2	12
[3,]	3	13

```
> rbind(a, b) # “row bind”
```

	[,1]	[,2]	[,3]
a	1	2	3
b	11	12	13

Матрицы

Присваивание имен столбцам и колонкам:

```
> m <- matrix(c(1:6), nrow=2, ncol=3)
```

```
> m
```

	[,1]	[,2]	[,3]
[1,]	1	3	5
[2,]	2	4	6

```
> rownames(m) <- c("r1", "r2")
```

```
> colnames(m) <- c("c1", "c2", "c3")
```

```
> m
```

	c1	c2	c3
r1	1	3	5
r2	2	4	6

Списки

- Список, в отличие от вектора, может хранить элементы разного типа
 - В частности, элементом может быть список

```
> L = list("A", c(1,2), 30)
```

Сравните:

```
> v=c(v, 4)
```

```
> L1 = list(L, 40)
```

```
#->list(list("A",c(1,2),30),40)
```

```
> L [[ 4 ]] = 4
```

```
> L [[ length( L ) + 1 ]] = 4
```

```
> L
```

```
[[1]]
```

```
[1] "A"
```

```
[[2]]
```

```
[1] 1 2
```

```
[[3]]
```

```
[1] 30
```

список
из 3
элементов

Каждый элемент –
вектор (в R 1 элемент
– вектор из самого себя)

Списки

- Как достать элемент списка

```
> L [ 3 ] #-> список (из 1 элемента)
```

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

Правило: одинарные скобки []
возвращают элемент того же типа,

```
> L [[ 3 ]]  
#элемент (= вектор из 1 элемента)
```

```
[1] 30
```

- Как достать элемент из списка списков?

```
> L=list(10,list(11,12,13), 20, list(30))
```

```
#L[[2]] - тоже список
```

```
> L[[2]][[3]]
```

```
[1] 13
```

СПИСКИ

Сравните с выдачей любого теста, например t.test

■ Именованные элементы

```
> L=list(10,20)
```

```
> L$abc=123
```

```
> L
```

```
[[1]]
```

```
[1] 10
```

```
[[2]]
```

```
[1] 20
```

```
$abc
```

```
[1] 123
```

```
> names(L)
```

```
[1] "" "" "abc"
```

```
> L[[3]]
```

```
[1] 123
```

```
> L$abc=123
```

```
[1] 123
```

```
> L$abc
```

```
[1] 123
```

```
> L[["abc"]]
```

```
[1] 123
```

Повторение: векторы, списки, матрицы и data frames

	элементы одного типа	элементы разных типов
1D	Вектор <pre>>v = c(1,2,3) >v = c(v, 4) >v [2] #-> вектор (из 1 элемента)</pre>	Список <pre>> L = list(1,2,3) > L1 = list(L, 4) #-> list(list(1,2,3), 4) > L [[length(L) + 1]] = 4 > L [2] #-> список (из 1 элемента) > L [[2]] #элемент (=вектор из 1 элемента)</pre>
2D	Матрица “2D вектор” <pre>>m=matrix(c(1,2,3,4), nrow=2) >m[,1] #колонка матрицы - вектор >m[2,] #ряд матрицы – вектор</pre>	data frame “список векторов (разных типов)” <pre>>df=data.frame(name=c("A", "B", "C"), grade=c(5,5,4)) >df[,1] #колонка d.f- вектор >df[1,] #ряд d.f- тоже data frame (как и d.f, является списком). Почему не вектор?</pre>
nD	array “n-мерный вектор”	



Элементарная статистика

Эксперимент:

как отличить «честную» монетку от «нечестной»?

Честная монетка: вероятность орла 0.5, вероятность решки 0.5

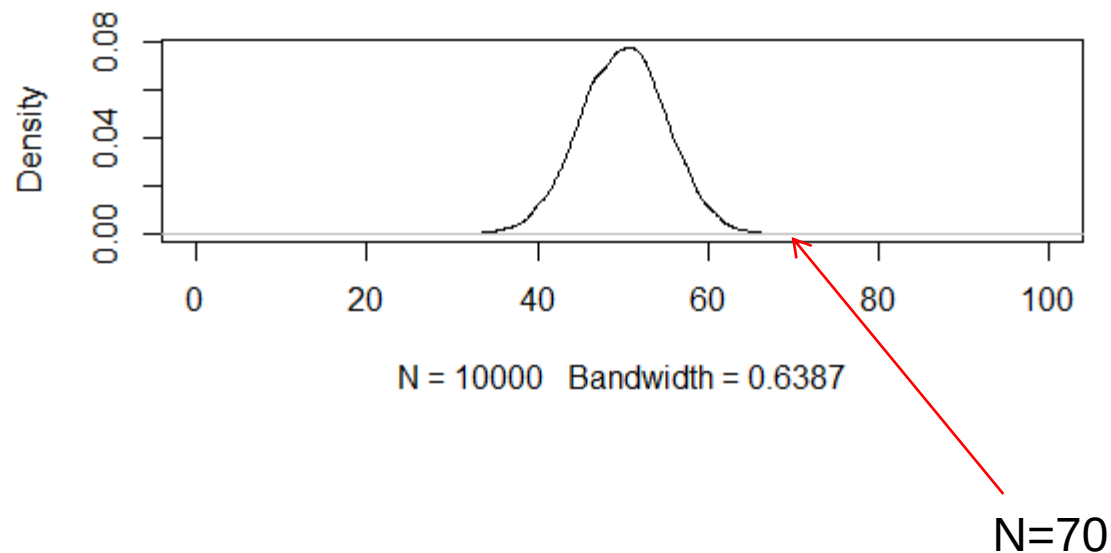
Нечестная монетка: вероятность орла 0.2, вероятность решки 0.8

Подбросим монетку 100 раз.

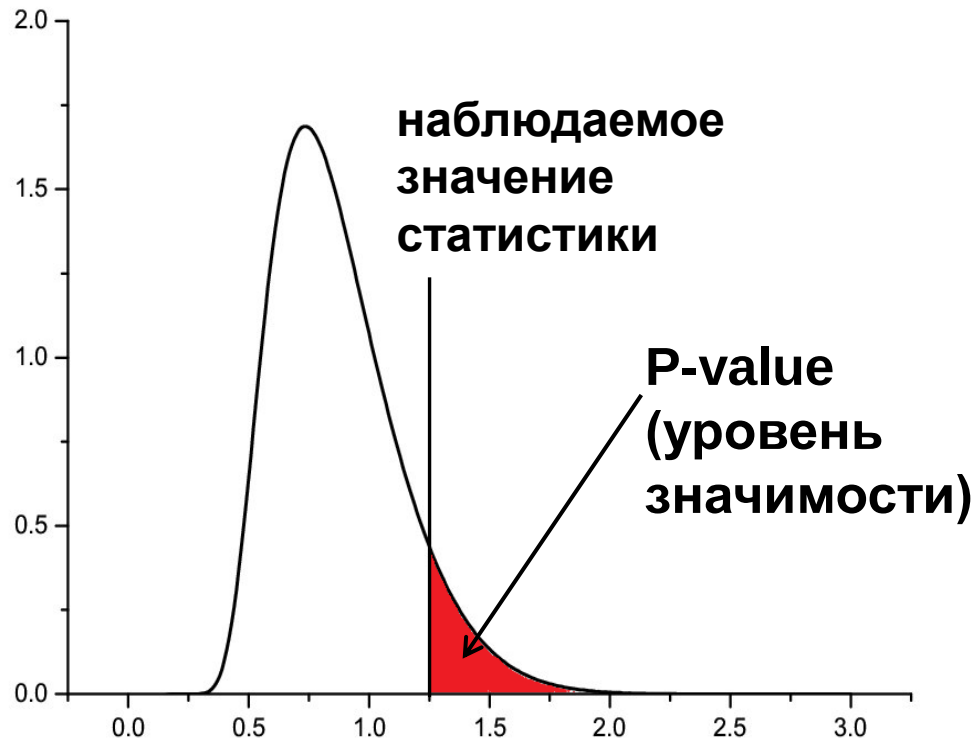
Решка выпала 70 раз. Какая у нас монетка?

Насколько можно быть уверенным в этом?

Распределение частот выпадения решки у честной монеты
(биномиальное распределение):



H_0 – нулевая гипотеза: мы кидали честную монету
 H_1 – альтернативная гипотеза: монета кривая



Что такое P-value?

- ✓ Вероятность наблюдаемого при нулевой гипотезе
- ✓ Вероятность ошибочно отвергнуть нулевую гипотезу (когда она верна)

Не строгие математические определения, главное – понять смысл!

Распределения – генерация

Распределение

- ✓ Нормальное распределение:
- ✓ `dnorm(x)` – плотность распределения
- ✓ `pnorm(q)` – функция распределения
- ✓ `qnorm(p)` – квантильная функция

Случайная генерация из распределения: `> set.seed(100)`

`> rnorm(5)`

`[1] 1.1568405 -0.8248219 0.1428891 -0.4784408 0.7561443`



Равномерное

```
runif(n, min=0, max=1)
```

```
> runif(5, 0, 1)
```

[1] 0.1972687 0.3090867 0.2865924 0.1409635 0.3441481

Биномиальное

```
rbinom(n, size, prob)
```

```
> rbinom(10, 100, 0.5)
```

[1] 54 47 55 50 47 45 52 45 58 52

Пуассона

```
rpois(n, lambda)
```

```
> rpois(10, 4)
```

[1] 2 3 2 4 10 3 2 3 5 6

Данные: вес цыплят в зависимости от рациона питания

```
> chick.w <- read.table("ChickWeight.tab", header=T)
```

```
> dim(chick.w)
```

```
[1] 20 2
```

```
> head(chick.w)
```

```
weight Diet
```

```
232      331      2
```

```
244      167      2
```

```
256      175      2
```

```
268       74      2
```

```
280      265      2
```

```
292      251      2
```

```
> tail(chick.w,3)
```

```
weight Diet
```

```
436      290      3
```

```
448      272      3
```

```
460      321      3
```

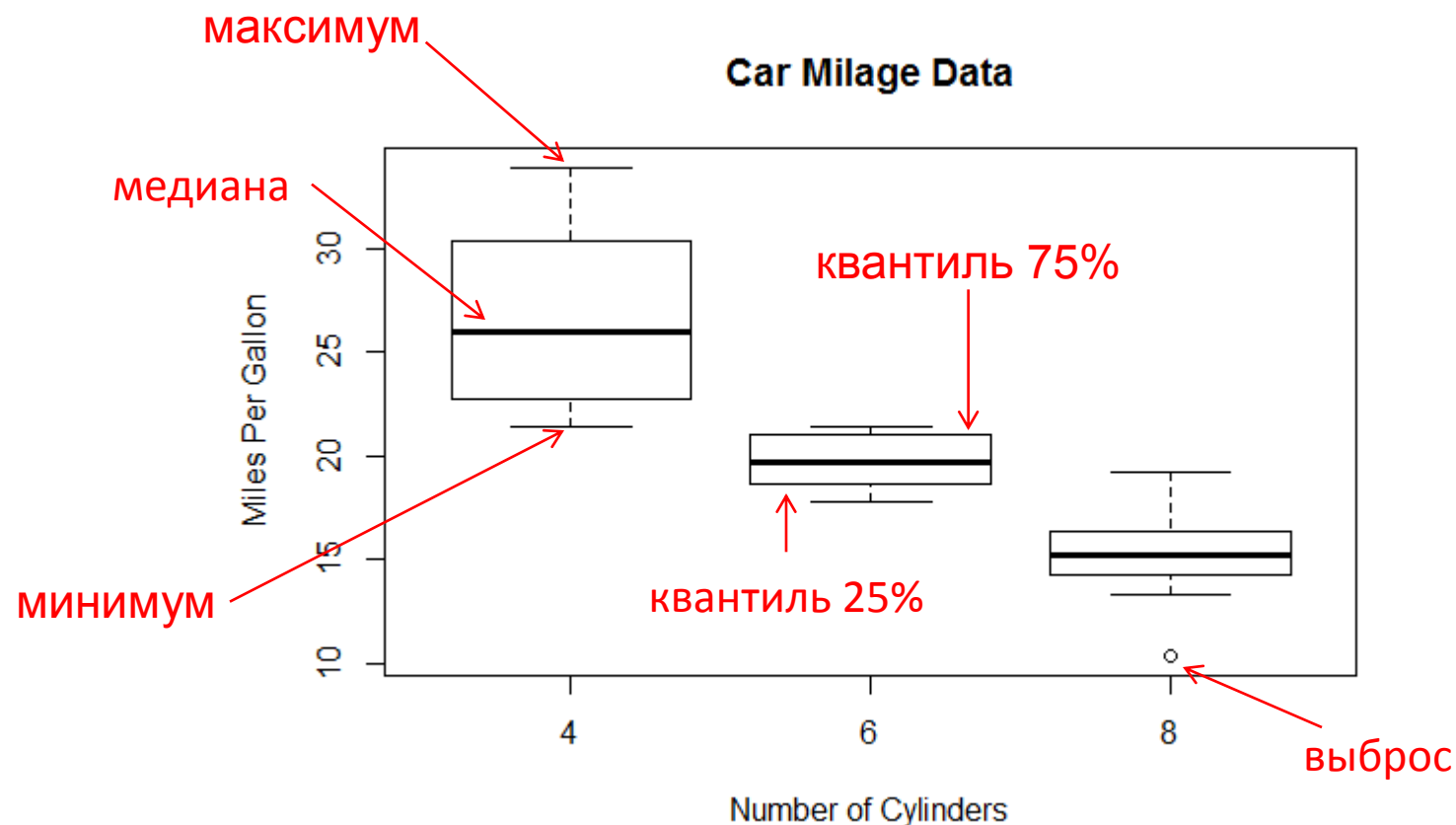
```
# weight – вес цыпленка (в граммах)
```

```
# Diet – тип рациона (2 или 3)
```

Задача:

понять, влияет ли рацион на вес

Boxplots

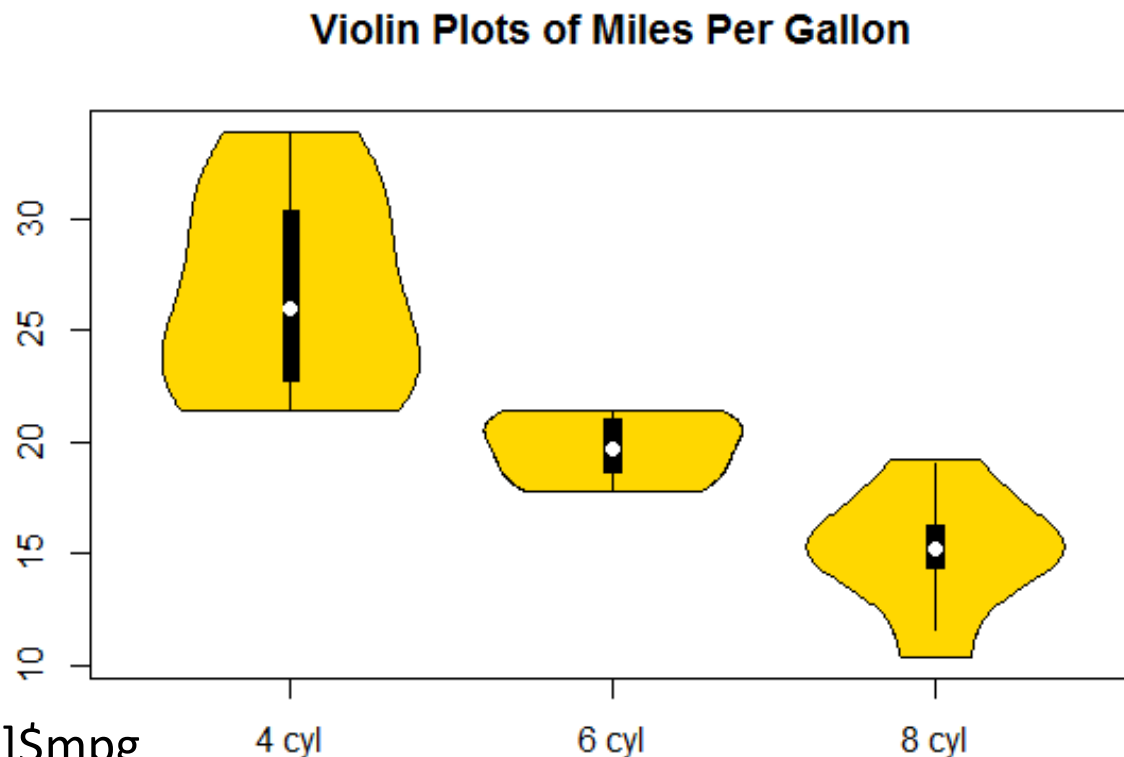


```
> boxplot(mpg~cyl,data=mtcars, main="Car Milage Data", xlab="Number of Cylinders",  
ylab="Miles Per Gallon")
```

Violin Plot: комбинация boxplot и графика плотности распределения

«The violin plot is like the lovechild between a density plot and a box-and-whisker plot.»

```
> library(vioplot)
> x1 <- mtcars[mtcars$cyl==4,]$mpg
> x2 <- mtcars[mtcars$cyl==6,]$mpg
> x3 <- mtcars[mtcars$cyl==8,]$mpg
> vioplot(x1, x2, x3, names=c("4 cyl", "6 cyl", "8 cyl"), col="gold")
title("Violin Plots of Miles Per Gallon")
```





Данные: вес цыплят в зависимости от рациона питания

```
> chick.w <- read.table("ChickWeight.tab", header=T)
```

```
> dim(chick.w)
```

```
[1] 20 2
```

```
> head(chick.w)
```

```
weight Diet
```

```
232      331      2
```

```
244      167      2
```

```
256      175      2
```

```
268       74      2
```

```
280      265      2
```

```
292      251      2
```

```
> tail(chick.w,3)
```

```
weight Diet
```

```
436      290      3
```

```
448      272      3
```

```
460      321      3
```

weight – вес цыпленка (в граммах)

Diet – тип рациона (2 или 3)

Задача:

понять, влияет ли рацион на вес

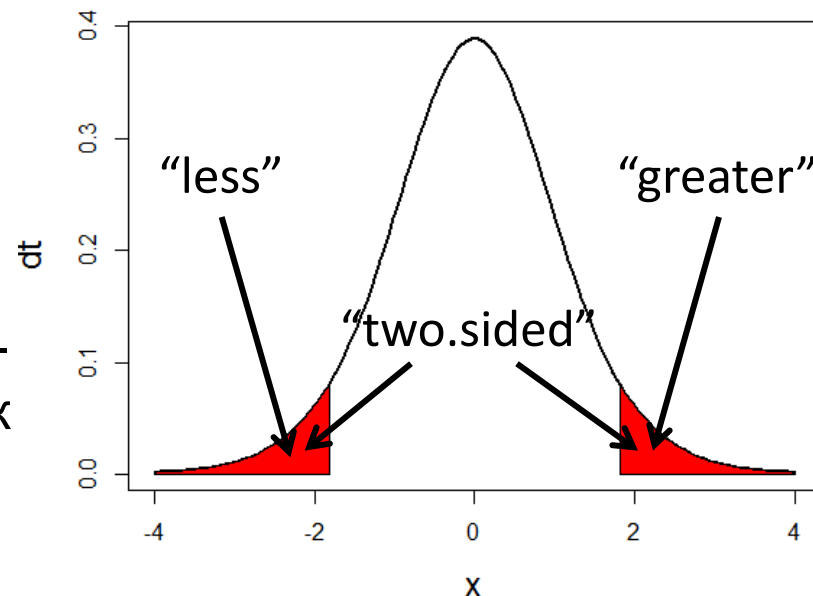
Сдвинуты ли выборки друг относительно друга?

	Выборки (почти) нормальные	Выборки совсем не нормальные
Выборки независимые	• t-тест – проверяет равенство средних	• U-критерий Манна-Уитни (Критерий суммы рангов Уилкоксона) • Kolmogorov-Smirnov test
Парные выборки	• Парный t-test – проверяет равенство разности случайных величин нулю	• Т-Критерий Вилкоксона

Student's (Gosset's) t-тест

- Введен Вильямом Госсетом в 1908 для оценки качества пива на пивоварне Guinness
- Используется для:
 - ☐ проверки равенства выборочного среднего заданному значению
 - ☐ проверки равенства средних значений двух серий измерений, сделанных для тех же объектов в разных условиях (например, состояние пациентов до и после лечения) – **paired t-test**
 - ☐ проверки равенства средних двух независимых выборок
- Предполагается, что случайные величины распределены **примерно нормально**
- При больших размерах выборок, распределение t-статистики приближается к нормальному

Распределение t-статистики



t-test для независимых выборок

Способ №1:

```
> chick.test <- t.test(w.diet.2, w.diet.3,
alternative="less")
```

Способ №2:

```
> chick.test <- t.test(chick.w$weight ~ chick.w$Diet,
alternative="less")
> chick.test$p.value
```

Welch Two Sample t-test

data: chick.w\$weight by chick.w\$Diet

t = -1.6588, df = 17.865, p-value = 0.05731

alternative hypothesis: true difference in means is less than 0

95 percent confidence interval:

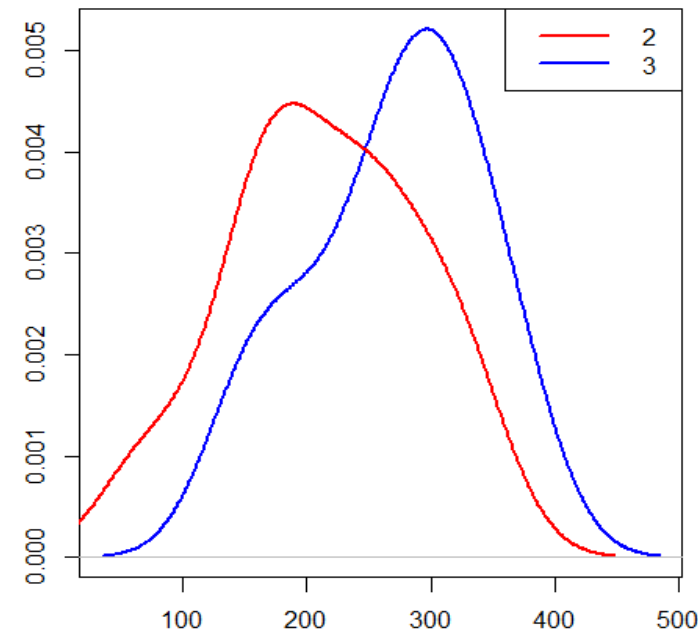
-Inf 2.548154

sample estimates:

mean in group 2 mean in group 3

214.7

270.3

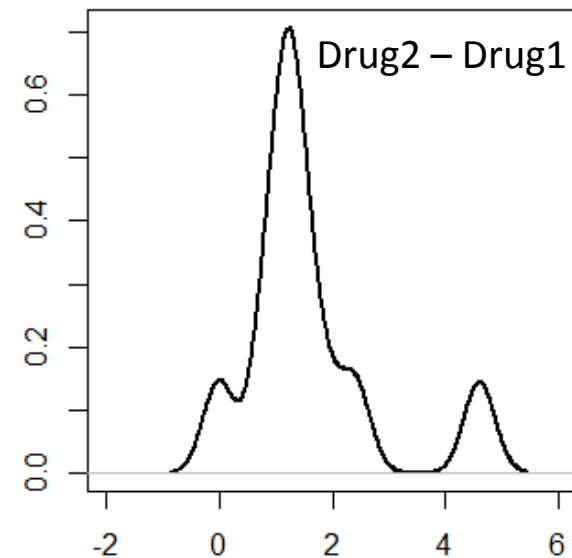
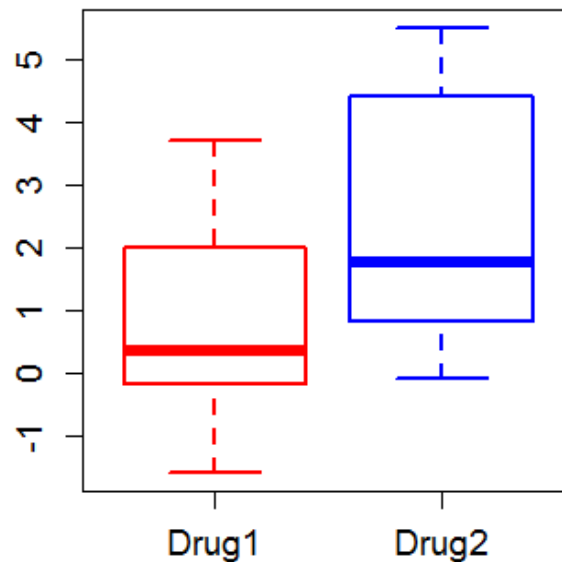


Данные: изменение длительности сна пациентов в зависимости от принимаемого лекарства

```
> sleep.paired <- read.table("sleep.paired.tab",header=T)
# ID – идентификатор пациента
# Drug1 и Drug2 – изменение длительности сна (в часах) при
# приеме лекарств 1 и 2
> sleep.paired
```

	ID	Drug1	Drug2
1	1	0.7	1.9
2	2	-1.6	0.8
3	3	-0.2	1.1
4	4	-1.2	0.1
5	5	-0.1	-0.1
6	6	3.4	4.4
7	7	3.7	5.5
8	8	0.8	1.6
9	9	0.0	4.6
10	10	2.0	3.4

Изменение длительности сна в зависимости от лекарства



Сдвинуты ли выборки друг относительно друга?

	Выборки (почти) нормальные	Выборки совсем не нормальные
Выборки независимые	• t-тест – проверяет равенство средних	• U-критерий Манна-Уитни (Критерий суммы рангов Уилкоксона) • Kolmogorov-Smirnov test
Парные выборки	• Парный t-test – проверяет равенство разности случайных величин нулю	• T-Критерий Вилкоксона

Парный t-тест

Помогло ли лекарство - стали ли пациенты дольше спать?

Способ №1:

```
> sleep.test <- t.test(sleep.paired$Drug1,  
+ sleep.paired$Drug2, paired=T, alternative="less")
```

Способ №2:

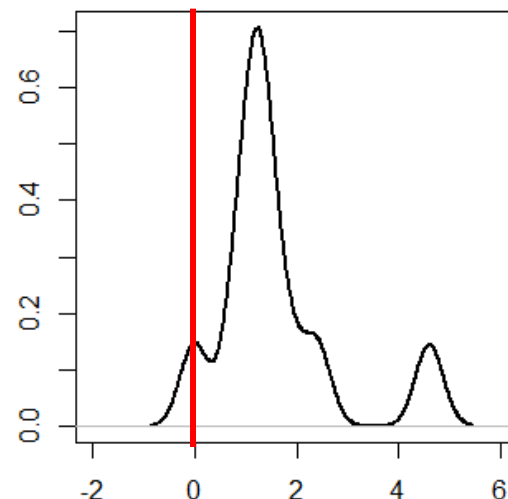
```
> diff <- sleep.paired$after - sleep.paired$before  
> t.test(diff) # объект htest
```

One Sample t-test

```
data: diff  
t = 4.0621, df = 9, p-value = 0.001416  
alternative hypothesis: true mean is greater than  
95 percent confidence interval:  
 0.8669947      Inf  
sample estimates:  
mean of x  
  1.58
```

Если забыть указать, что тест парный:

```
> t.test(sleep.paired$Drug1, sleep.paired$Drug2,  
alternative="less")$p.value  
[1] 0.03969707
```





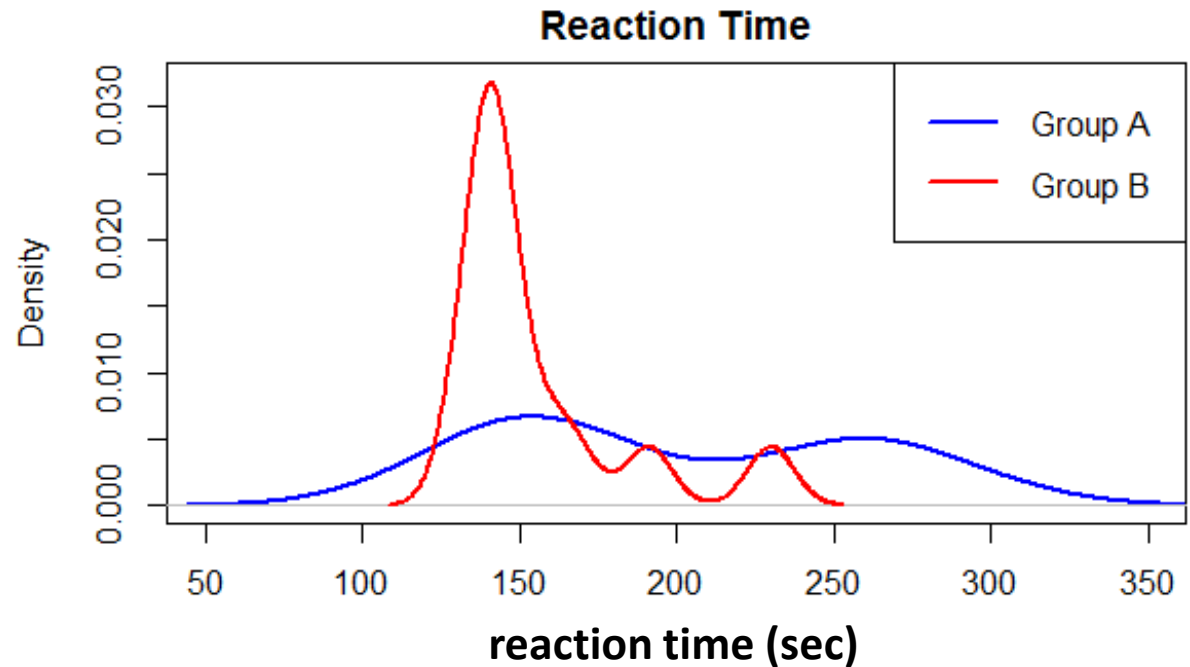
Непараметрические тесты

Данные: влияние анестетика на время реакции пациентов на световой раздражитель

```
rt <- read.table("anaesthetic.reaction.time.tab",  
sep="\t", header=T)
```

Mean.RT – среднее время реакции; Group: A/B – with/without anesthetic

```
> head(rt)  
  Mean.RT Group  
1      131     B  
2      135     A  
3      138     B  
4      138     B  
5      139     A  
6      141     B
```



Больше ли время реакции у пациентов под воздействием анестетика?

Сдвинуты ли выборки друг относительно друга?

	Выборки (почти) нормальные	Выборки совсем не нормальные
Выборки независимые	• t-тест – проверяет равенство средних	• U-критерий Манна-Уитни (Критерий суммы рангов Уилкоксона) • Kolmogorov-Smirnov test
Парные выборки	• Парный t-test – проверяет равенство разности случайных величин нулю	• T-Критерий Вилкоксона

U-критерий Манна-Уитни

- Используется для тестирования гипотезы, что значения в одной из выборок в среднем (стохастически) больше, чем в другой
- H_0 : выборки не отличаются
- Позволяет выявлять различия в значении параметра между малыми выборками
- При больших размерах выборок, распределение U-статистики приближается к нормальному

U-критерий Манна-Уитни

Способ №1:

```
> wilcox.test(rt$Mean.RT~rt$Group, alternative="greater")
```

wilcoxon rank sum test with continuity correction

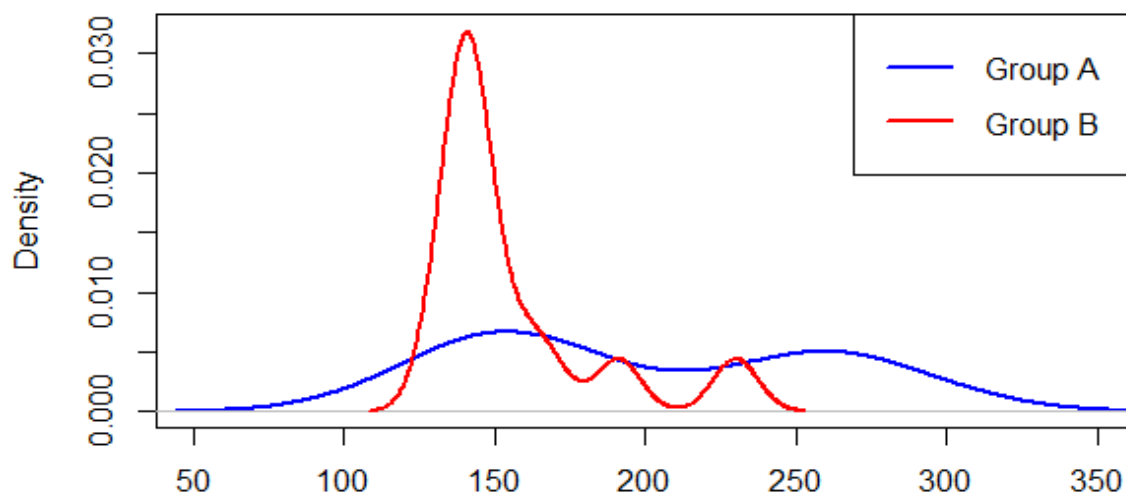
data: rt\$Mean.RT by rt\$Group

W = 126, **p-value = 0.01633**

alternative hypothesis: true location shift is greater than 0

Способ №2:

```
> wilcox.test(rt[rt$Group=="A", "Mean.RT."],  
rt[rt$Group=="B", "Mean.RT."], alternative="greater")
```

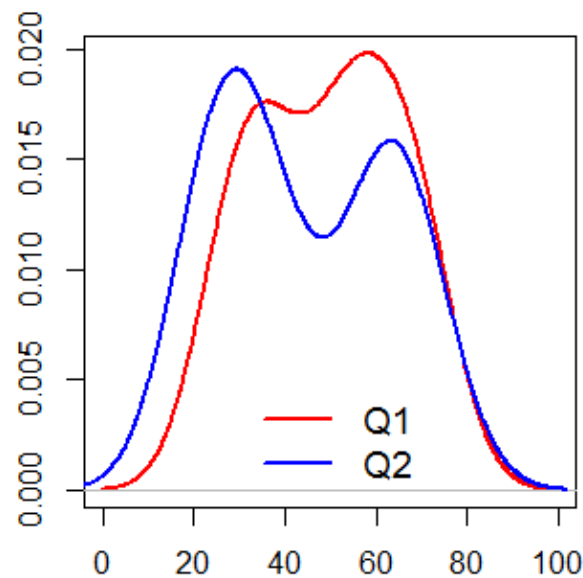


Данные: ответы студентов на вопросы теста

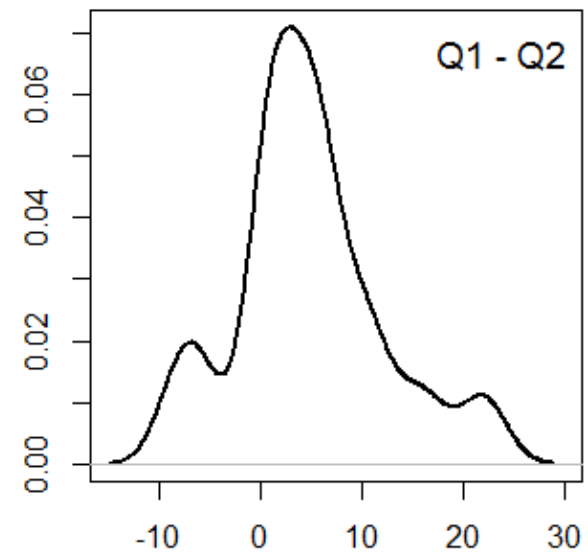
```
> test <- read.csv("StudentTest.csv")  
# Student – студент, отвечающий на вопрос  
# Q1, Q2 – баллы (от 0 до 100) за 1 и 2 вопросы
```

```
> head(test)  
Student Q1 Q2  
1      1 78 67  
2      2 24 24  
3      3 64 62  
4      4 55 58  
5      5 74 28  
6      6 52 36
```

Распределения баллов
за вопросы



Распределения разницы
баллов за вопросы



Предположение: студенты лучше отвечали на первый вопрос.

Сдвинуты ли выборки друг относительно друга?

	Выборки (почти) нормальные	Выборки совсем не нормальные
Выборки независимые	• t-тест – проверяет равенство средних	• U-критерий Манна-Уитни (Критерий суммы рангов Уилкоксона) • Kolmogorov-Smirnov test
Парные выборки	• Парный t-test – проверяет равенство разности случайных величин нулю	• T-Критерий Вилкоксона

T-Критерий Вилкоксона

Способ №1:

```
> wilcox.test(test$Q1, test$Q2, paired=T,  
alternative="greater", exact=F)
```

wilcoxon signed rank test with continuity correction

data: test\$Q1 and test\$Q2

$V = 114.5$, $p\text{-value} = 0.008503$

alternative hypothesis: true location shift is greater than 0

Способ №2:

```
> wilcox.test(test$Q1-test$Q2, alternative="greater",  
exact=F)
```

Предположение, что студенты лучше отвечали на первый вопрос, подтвердилось на 1% уровне значимости.

Элементарная статистика: типичные вопросы

- Как распределены наблюдения в выборке?
 - является ли выборка нормальной? (`ks.test`, `shapiro.test`)
- Сравнение двух выборок:
 - из одного ли они распределения? (`ks.test`)
 - сдвину-ты ли они друг относительно друга? (`t.test`, `wilcox.test`, `ks.test`)



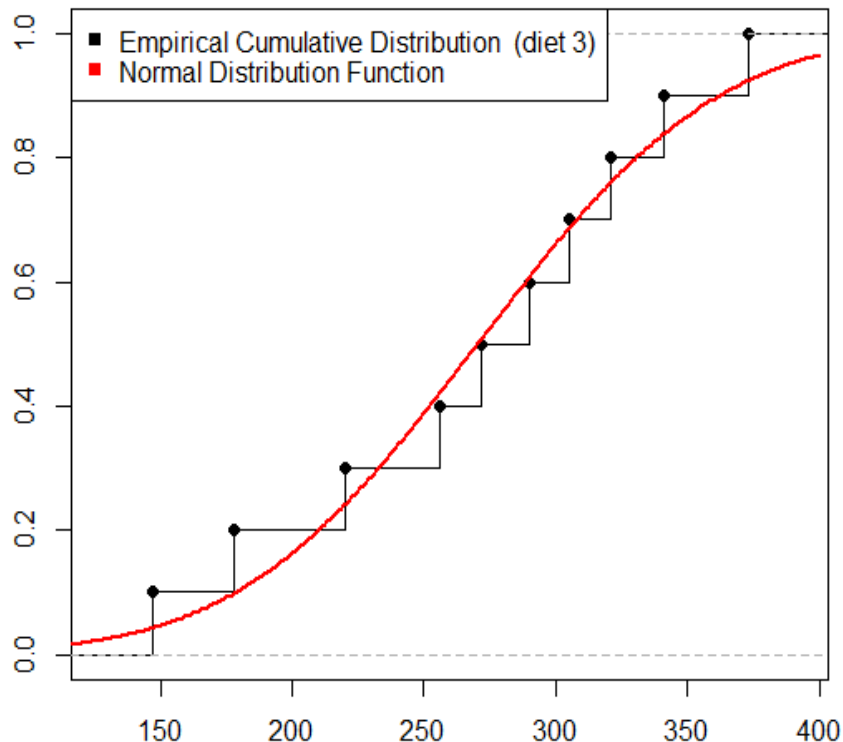
(*) Тесты на нормальность распределения

Статистические тесты для сравнения распределений

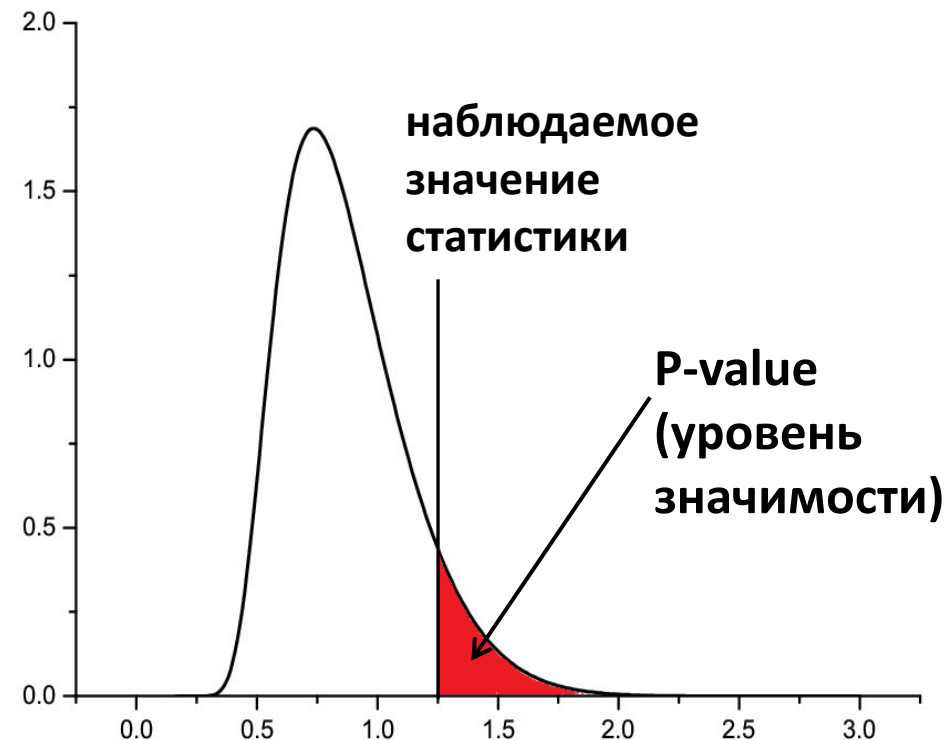
Тест Колмогорова-Смирнова:

- чувствителен к отличиям в форме распределений и их сдвигу относительно друг друга
- H_0 : распределения совпадают
- **плохо работает на маленьких выборках**
- применим только для непрерывных распределений

Сравнение эмпирического и теоретического распределений



Распределение статистики при нулевой гипотезе



Статистические тесты для сравнения распределений

Сравнение эмпирического распределения с теоретическим:

тест на нормальность

```
> ks.test(w.diet.3, "pnorm", mean(w.diet.3), sd(w.diet.3))
```

One-sample Kolmogorov-Smirnov test

data: w.diet.3

D = 0.1209, p-value = 0.9944

alternative hypothesis: two-sided

Сравнение распределений двух выборок:

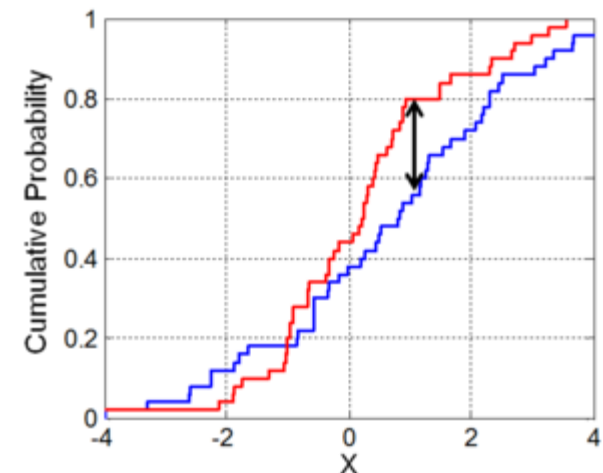
```
> ks.test(w.diet.2, w.diet.3)
```

Two-sample Kolmogorov-Smirnov test

data: w.diet.2 and w.diet.3

D = 0.4, p-value = 0.4175

alternative hypothesis: two-sided



Объект класса *htest*

Многие статистические тесты в R возвращают объект класса *htest*:

```
> diet3.ks <- ks.test(w.diet.2,w.diet.3)
> diet3.ks
```

Two-sample Kolmogorov-Smirnov test

data: w.diet.2 and w.diet.3

D = 0.4, p-value = 0.4175

alternative hypothesis: two-sided

```
> class(diet3.ks)
```

```
[1] "htest"
```

```
> names(diet3.ks)
```

```
[1] "statistic" "p.value" "alternative" "method"
```

```
[5] "data.name"
```

```
> diet3.ks$statistic
```

```
D
```

```
0.4
```

```
> diet3.ks$p.value
```

```
[1] 0.4175
```

Статистические тесты для сравнения распределений

Тест Shapiro-Wilk:

- проверяет гипотезу, что выборка пришла из **нормального распределения**
- H_0 : выборка является нормальной
- мощнее, чем тест Колмогорова-Смирнова (то есть с меньшей вероятностью ошибочно принимает H_0)
- размер выборки от 3 до 5000

```
> shapiro.test(w.diet.3) # возвращает объект htest
```

Shapiro-wilk normality test

data: w.diet.3

w = 0.9705, **p-value = 0.895**

```
> shapiro.test(w.diet.2)$p.value  
[1] 0.948785
```