



ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ
УПРАВЛЕНИЯ

Основан в 1919 году

Обработка данных в среде офисных приложений: введение в R

Светлана Андреевна Суязова (Аксюк)

sa_aksyuk@guu.ru

4 сентября 2021

Занятие 1

- "Введение в язык статистической обработки данных R"
- три встречи, 5 пар
- сквозное задание + тест (на последней паре)





R - это

 язык статистической обработки и визуализации данных

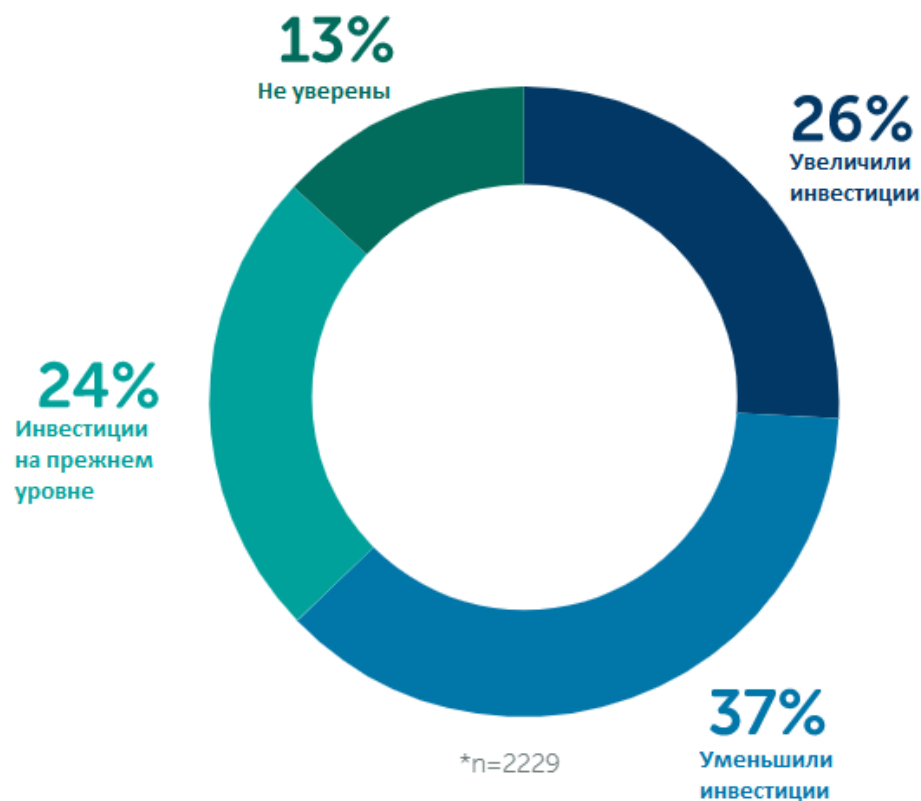
 интерфейсы для чтения и сбора данных

 постоянно пополняемая база пакетов

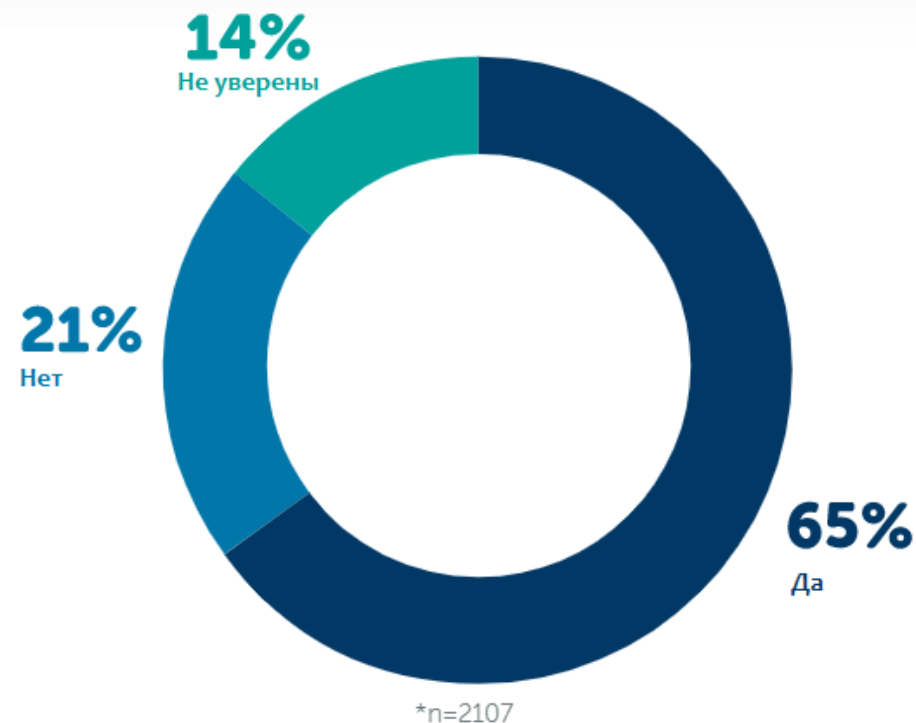
 инструменты для создания отчётов, презентаций, веб-приложений (`knitr`, `shiny`, `xaringan`)

 IDE на выбор: RStudio, Vim, Emacs + ESS, Jupyter Notebook, Revolution R Enterprise, etc.

Оказал ли COVID-19 влияние на инвестиции в аналитику данных внутри вашей организации?

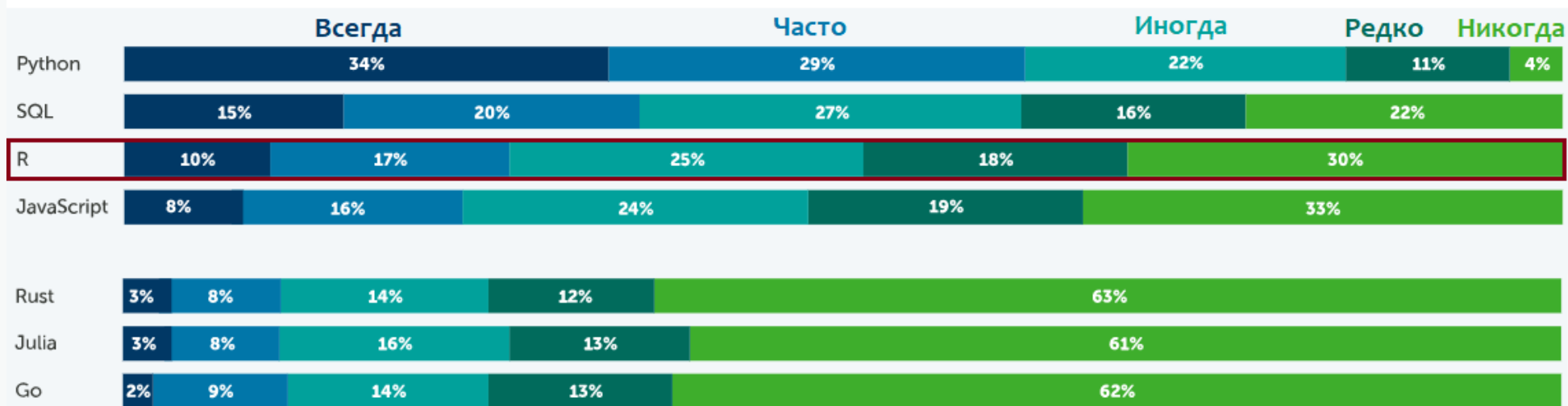


Поощряет ли ваш работодатель использование инструментов с открытым исходным кодом (open-source)?



Источник: [Онлайн опрос на платформе Anaconda.com](#), апрель-май 2021 года

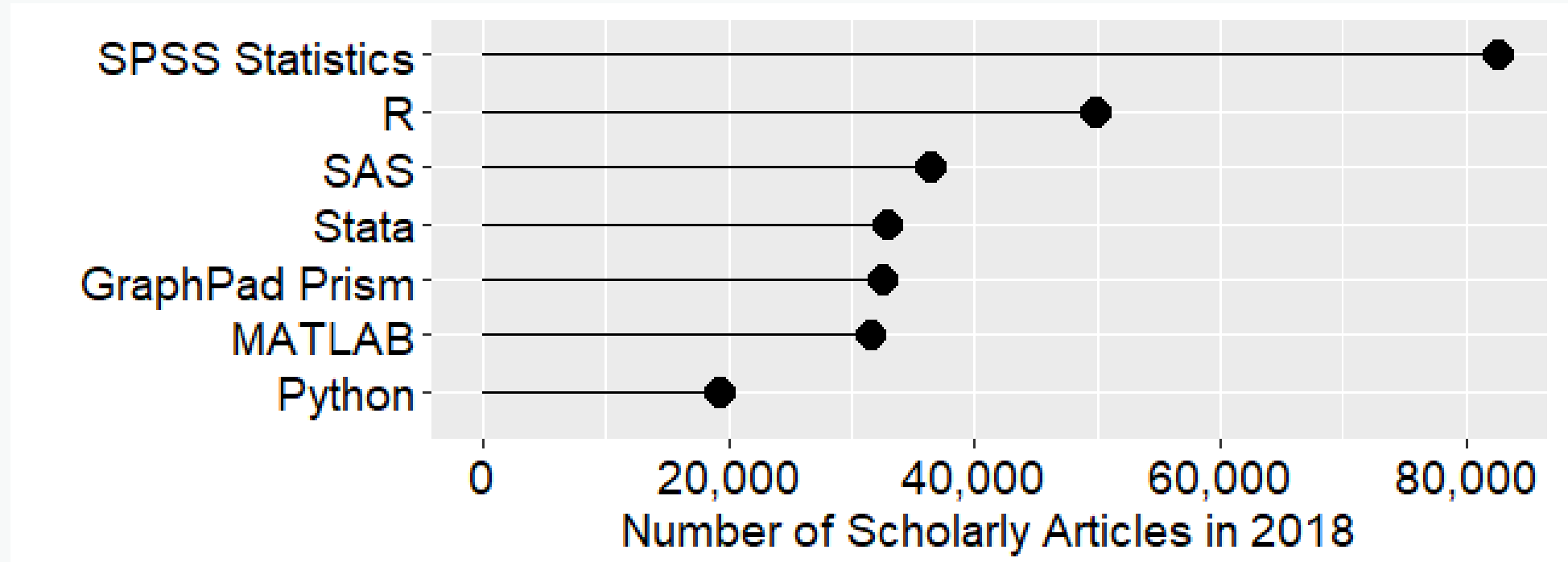
Как часто в анализе данных вы используете следующие языки?



*n=3104

Источник: [Онлайн опрос на платформе Anaconda.com](#), апрель-май 2021 года

В 2018 гг. R на втором месте по использованию в академических исследованиях (упоминания в Google Scholar)



Источник: [The Popularity of Data Science Software, r4stats.com](https://r4stats.com/)

Содержание курса

- типы и структуры данных, синтаксис, интерфейс RStudio
- графические системы R
- предварительный анализ данных (кросс-секционная выборка) и построение линейных регрессионных моделей
- интерактивные отчёты на Rmarkdown, экспорт отчётов в html и word

Материалы

- Суязова С.А. Введение в язык статистической обработки данных R: учебное пособие. – М.: ГУУ, 2018. pdf
- Репозиторий со скриптами и pdf-файлами к лекциям и лабораторным: github.com/aksyuk/R_data_glimpse
- А.Б. Шипунов, Е.М. Балдин, П.А. Волкова, А.И. Коробейников, С.А. Назарова, С.В. Петров, В.Г. Суфиянов Наглядная статистика. Используем R! – 2020, исправленная версия. pdf
- Роберт И. Кабаков R в действии. Анализ и визуализация данных в программе R. – ДМК Пресс, 2014. pdf

Лекция №1: основы

Особенности работы в R:

- ☒ функции в основе всего
- ☒ все объекты сессии хранятся в оперативной памяти
- ☒ по умолчанию параллельные вычисления не поддерживаются
- ☒ текстовые результаты выводятся в консоль
- ☒ графические результаты выводятся во встроенный браузер, либо в файл

Типы переменных

- числовой, целый: `integer`
- числовой, с плавающей **точкой**: `double` (по умолчанию)

```
typeof(42)
```

```
## [1] "double"
```

```
is.integer(42)
```

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

```
typeof(integer(42))
```

```
## [1] "integer"
```

```
typeof(6.5)
```

```
## [1] "double"
```

Типы переменных

- текстовый: character (любые парные кавычки)

```
typeof("Введение в язык R")
```

```
## [1] "character"
```

```
typeof('Курс "Введение в язык R"')
```

```
## [1] "character"
```

```
typeof('42')
```

```
## [1] "character"
```

Типы переменных

- логический: `logical`

F равносильно FALSE, T равносильно TRUE

```
2 == '2'
```

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

```
'осень' != 'лето'
```

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

Структуры данных

Вектор – одномерный набор элементов одного типа, нумерация элементов с единицы. Функция `c()` – от **c**oncatenate.

```
x <- c(1, 1, 2, 3)
typeof(x)
```

```
## [1] "double"
```

```
y <- c(TRUE, 1, '2', "3")
typeof(y)
```

```
## [1] "character"
```

```
x == y
```

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

Структуры данных

Матрица – вектор с двумя измерениями.

```
x.matrix <- matrix(c(1, 2, 3, 4, 5, 6), 2, 3)
```

```
x.matrix
```

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

```
dim(x.matrix)
```

```
## [1] 2 3
```

Структуры данных

Список – одномерный набор элементов, типы м.б. разными

```
my.list <- list(index.name = "ВВП РФ, трлн долл.",  
               year = 2017:2020, value = c(1.58, 1.66, 1.69, 1.48))  
my.list
```

```
## $index.name  
## [1] "ВВП РФ, трлн долл."  
##  
## $year  
## [1] 2017 2018 2019 2020  
##  
## $value  
## [1] 1.58 1.66 1.69 1.48
```

Структуры данных

Фрейм данных (`data.frame`) – таблица с показателями-столбцами (векторы) и наблюдениями-строками, **список векторов**.

```
df.2019 <- data.frame(city = c("Москва", "Воронеж",  
                                "Липецк"),  
                      popul.mln = c(12.5, 1.1, 0.6),  
                      area.sq.km = c(2561.5, 596.51, 330.15)  
df.2019
```

##		city	popul.mln	area.sq.km
## 1		Москва	12.5	2561.50
## 2		Воронеж	1.1	596.51
## 3		Липецк	0.6	330.15

Структуры данных: фрейм

```
str(df.2019)          # посмотреть структуру любого объекта
```

```
## 'data.frame':    3 obs. of  3 variables:  
##  $ city          : chr  "Москва" "Воронеж" "Липецк"  
##  $ popul.mln     : num  12.5  1.1  0.6  
##  $ area.sq.km    : num  2562  597  330
```

```
colnames(df.2019)     # посмотреть названия столбцов фрейма
```

```
## [1] "city"          "popul.mln"    "area.sq.km"
```

Базовые операторы

Присваивание справа налево: <-, сочетание **Alt + "-"**

Выбор элемента: []

Выбор элемента списка: \$

```
df.2019$city
```

```
## [1] "Москва" "Воронеж" "Липецк"
```

```
df.2019$city[1]
```

```
## [1] "Москва"
```

Базовые операторы: []

```
df.2019[1, ] # выбрать первую строку фрейма
```

```
##      city popul.mln area.sq.km  
## 1 Москва      12.5    2561.5
```

```
df.2019[, 1] # выбрать первый столбец фрейма
```

```
## [1] "Москва" "Воронеж" "Липецк"
```

```
df.2019[1, 2] # выбрать элемент из строки 1, столбца 2
```

```
## [1] 12.5
```

Базовые операторы: []

```
df.2019[, 2:3] # выбрать второй и третий столбцы фрейма
```

```
##      popul.mln area.sq.km
## 1         12.5    2561.50
## 2          1.1     596.51
## 3          0.6     330.15
```

```
df.2019[, c('popul.mln', 'area.sq.km')] #по именам столбцов
```

```
##      popul.mln area.sq.km
## 1         12.5    2561.50
## 2          1.1     596.51
## 3          0.6     330.15
```

Ключевые приёмы: векторизация

Векторизация – применение функции (оператора) ко всем элементам вектора

```
# проверить, в каких городах население больше миллиона  
df.2019$popul.mln > 1
```

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

```
# названия этих городов  
df.2019$city[df.2019$popul.mln > 1]
```

```
## [1] "Москва" "Воронеж"
```

Ключевые приёмы: векторизация

```
x <- c(1, -2, 3, -4)
# увеличить все элементы вектора на 1
y <- x + 1
# вывести результат
y
```

```
## [1] 2 -1 4 -3
```

```
# увеличить только положительные элементы вектора на 2
x[x > 0] <- x[x > 0] + 2
# вывести результат
x
```

```
## [1] 3 -2 5 -4
```

Ключевые приёмы: apply-функции

- Семейство `apply()`: применение функции к элементам списка

```
apply(df.2019[, 2:3], 2, mean) # средние насел. и площадь
```

```
##      popul.mln  area.sq.km  
##      4.733333 1162.720000
```

```
sapply(df.2019[, 2:3], mean) # применяем то же к списку
```

```
##      popul.mln  area.sq.km  
##      4.733333 1162.720000
```

Ключевые приёмы: apply-функции

```
lapply(df.2019, mean)
```

```
## Warning in mean.default(X[[i]], ...): аргумент не является числом  
## возвращаю NA
```

```
## $city  
## [1] NA  
##  
## $popul.mln  
## [1] 4.733333  
##  
## $area.sq.km  
## [1] 1162.72
```


Ключевые приёмы: вызов справки

- **Работа с документацией**

```
# вызов справки по функции apply  
?apply
```

```
# поиск по сайту проекта (открывается в браузере)  
RSiteSearch('apply')
```

около 400 тысяч вопросов с тегом [r] на [stackoverflow.com](https://stackoverflow.com/questions/tagged/r):
stackoverflow.com/questions/tagged/r

Распространённые ошибки

```
c(1, 2, 3, 4) # функция c() написана по-русски
```

Error in c(1, 2, 3, 4) : could not find function "c"

```
sum(df.2019$city) # пытаемся посчитать что-то нечисловое
```

Error in sum(df.2019\$city) : invalid 'type'
(character) of argument

```
df.2019$city
```

```
## [1] "Москва" "Воронеж" "Липецк"
```

Подробнее

- Главы 1-2 учебного пособия "Введение в язык статистической обработки данных R", практические примеры.

Лабораторная работа №1

- Познакомимся с интерфейсом R Studio, научимся импортировать данные из .csv и с помощью API сайтов

