

Язык R - операторы

2+2	сумма
5-3	вычитание
2*4	умножение
5/3	деление
(5-3)*2	последовательность действий работает как ожидается

Переменные

<code>x <- 2</code>	запись значения 2 в переменную x
<code>x = 2</code>	то же самое
<code>x = "пер еме нна я"</code>	все в кавычках интерпретируется как последовательность символов (string)
<code>x=c(1,2,3)</code>	<code>c()</code> создает вектор (упорядоченный набор значений)
<code>x[1]</code>	<code>[]</code> используются для индексации переменных. <code>x[1]</code> достает первый элемент из вектора x
<code>x[1,1]</code>	двумерные объекты (матрицы, таблицы) индексируются через [индекс строки.индекс столбца]

Переменные (cont)

<code>x\$column1</code>	данные столбцов в таблицах (data frame) можно получать по названию столбца через \$. В этом случае столбец возвращается как вектор значений. \$ также используется, чтобы индексировать любые названные переменные из сложных объектов (таблицы, листы, листы с листами и пр.)
-------------------------	--

Функции

<code>sum(1,2,3)</code>	сумма
<code>sqrt(16)</code>	квадратный корень
<code>c(1,2,3)</code>	совмещение трех значений в вектор
<code>paste("Hello","world",sep=" ")</code>	совмещение нескольких текстов в одно значение
<code>unique(c(1,1,2))</code>	функции в функциях. <code>unique()</code> возвращает только уникальные значения из вектора

Общая структура функции: `function(x, arg1=0, arg2=TRUE)`. Где x - это вводные данные, а arg1, arg2 - это аргументы, контролирующие поведение функции. Иногда ввод не требуется и задан автоматически: `stylo(gui =FALSE)`



By ashel
cheatography.com/ashel/

Not published yet.
Last updated 28th June, 2020.
Page 1 of 2.

Sponsored by [Readable.com](https://readable.com)
Measure your website readability!
<https://readable.com>

Язык R - рабочие директории

`getwd()` показывает путь к папке, которую R считает рабочей в данный момент

`setwd("path/of/old")` задает рабочую директорию по указанному пути

Рабочая директория - это папка, которую работающий R использует для референции. Файлы с кодом и внешние файлы, которые будут нужны в работе лучше держать внутри рабочей директории, чтобы к ним легко было получить доступ. Новые файлы также сохраняются туда

RStudio

File -> New Project создает новую папку с проектом, которую автоматически делает рабочей.

File -> Open Project открывает существующий проект и переключает рабочую папку.

Рабочую папку можно изменить вручную из RStudio из панели **Files**, которая находится снизу справа и используется для навигации по файловой системе (отображает файлы в рабочей папке по умолчанию). Переместитесь в нужную папку и нажмите **More -> Set As Working Directory**

R - библиотеки

`install.packages()` устанавливает библиотеку из репозитория CRAN

`library()` загружает библиотеку в память R

`install.packages("stylo")` скажет и установит пакет `stylo()` на компьютер

`library(stylo)` загрузит существующую библиотеку в память R и позволит работать с ее функциями.

R - помощь

`help(stylo)` вызывает документацию

`?stylo()` --/--

Любая функция в R снабжена документацией.

Пакет stylo

`stylo()` дистанции

`classify()` машинное обучение

`rolling.classify()` МО скользящее по тексту

`oppose()` ключевые слова

`imposters()` верификация авторства

Основные функции пакета stylo

Функции и директории

stylo() ищет тексты в папке **corpus**, которая должна находиться в рабочей директории

classify() ищет тренировочный корпус в папке **primary_set**, а проверочный в **secondary_set**

rolling.classify() ищет тренировочный корпус в папке **reference_set**, а текст для скользящей классификации - в **test_set**

oppose() сравнивает тексты из двух папок **primary_set** и **secondary_set**

imposters() работает напрямую с таблицами частот и поэтому требует отдельной подготовки данных

NB. **oppose()** по умолчанию считает язык корпуса английским, поэтому эту функцию надо запускать с аргументом: `oppose(corporus.language = "Other")`

Классы

Все функции в **stylo** считывают классы из названий файлов.

Символы перед первой нижней чертой задают класс текста (*Dostoevsky_1846.txt* и *Dostoevsky_1864.txt*, будут отнесены к одному классу *Dostoevsky*).

Классы нужны для цветового кодирования результатов (все тексты из класса *Dostoevsky* будут раскрашены одним цветом) и проверки результатов машинной классификации.

Запуск функции "вручную"

```
stylo(gui = FALSE,
      corpus.language = "Other",
      analyzed.features = "c",
      ngram.size = 4,
      mfw.min=400,
      mfw.max=400,
      analysis.type = "CA")
```