

| | |
|---|----|
| 3.6. Отображение загруженных пакетов через путь поиска | 75 |
| 3.7. Просмотр списка установленных пакетов | 76 |
| 3.8. Доступ к функциям в пакете | 77 |
| 3.9. Доступ к встроенным наборам данных | 78 |
| 3.10. Установка пакетов из CRAN | 80 |
| 3.11. Установка пакета из GitHub | 81 |
| 3.12. Установка или изменение зеркала CRAN по умолчанию | 82 |
| 3.13. Запуск сценария | 84 |
| 3.14. Запуск пакетного сценария | 85 |
| 3.15. Поиск домашнего каталога R | 87 |
| 3.16. Настройка запуска R | 88 |
| 3.17. Использование R и RStudio в облаке | 91 |

Глава 4. Ввод и вывод

93

| | |
|--|-----|
| 4.1. Ввод данных с клавиатуры | 93 |
| 4.2. Вывод меньшего числа цифр (или большего) | 94 |
| 4.3. Перенаправление вывода в файл | 96 |
| 4.4. Список файлов | 97 |
| 4.5. Не удается открыть файл в Windows – что делать? | 99 |
| 4.6. Чтение записей фиксированной ширины | 100 |
| 4.7. Чтение файлов табличных данных | 102 |
| 4.8. Чтение из файлов CSV | 105 |
| 4.9. Запись в файлы CSV | 107 |
| 4.10. Чтение табличных или CSV-данных из интернета | 109 |
| 4.11. Чтение данных из Excel | 110 |
| 4.12. Запись таблицы данных в Excel | 111 |
| 4.13. Чтение данных из файла SAS | 113 |
| 4.14. Чтение данных из таблиц HTML | 115 |
| 4.15. Чтение файлов со сложной структурой | 117 |
| 4.16. Чтение из баз данных MySQL | 121 |
| 4.17. Доступ к базе данных с помощью dbplyr | 124 |
| 4.18. Сохранение и транспортировка объектов | 125 |

Глава 5. Структуры данных

128

| | |
|---|-----|
| 5.1. Добавление данных в вектор | 135 |
| 5.2. Вставка данных в вектор | 136 |
| 5.3. Правило повторного использования | 137 |
| 5.4. Создание фактора (категориальной переменной) | 139 |
| 5.5. Объединение нескольких векторов в один вектор и фактор | 140 |
| 5.6. Создание списка | 141 |
| 5.7. Выбор элементов по месту в списке | 143 |
| 5.8. Выбор элементов списка по имени | 144 |
| 5.9. Создание ассоциативного списка «имя/значение» | 146 |
| 5.10. Удаление элемента из списка | 147 |
| 5.11. Преобразование списка в вектор | 148 |
| 5.12. Удаление элементов NULL из списка | 149 |
| 5.13. Удаление элементов списка с использованием условия | 150 |
| 5.14. Инициализация матрицы | 152 |
| 5.15. Операции с матрицами | 153 |
| 5.16. Задание описательных имен для строк и столбцов матрицы | 154 |

| | |
|--|------------|
| 5.17. Выбор одной строки или столбца из матрицы..... | 155 |
| 5.18. Инициализация таблицы данных из данных столбца..... | 156 |
| 5.19. Инициализация таблицы данных из данных строки | 157 |
| 5.20. Добавление строк в таблицу данных | 160 |
| 5.21. Выбор столбцов таблицы по их месту в таблице данных..... | 162 |
| 5.22. Выбор столбцов таблицы данных по имени | 165 |
| 5.23. Изменение имен столбцов таблицы данных | 167 |
| 5.24. Удаление значений NA из таблицы данных | 168 |
| 5.25. Исключение столбцов по имени | 169 |
| 5.26. Объединение двух таблиц данных..... | 170 |
| 5.27. Объединение таблиц данных по общему столбцу | 171 |
| 5.28. Приведение атомарных типов данных..... | 173 |
| 5.29. Приведение структурированных типов данных | 174 |
| Глава 6. Преобразование данных | 177 |
| 6.1. Применение функции ко всем элементам списка | 177 |
| 6.2. Применение функции к каждой строке таблицы данных..... | 179 |
| 6.3. Применение функции к каждой строке матрицы | 180 |
| 6.4. Применение функции к каждому столбцу | 182 |
| 6.5. Применение скалярной функции к векторам или спискам | 183 |
| 6.6. Применение функции к группам данных | 186 |
| 6.7. Создание нового столбца по условию | 187 |
| Глава 7. Строки и даты | 189 |
| 7.1. Получение длины строки..... | 191 |
| 7.2. Конкатенация строк | 191 |
| 7.3. Извлечение подстрок | 192 |
| 7.4. Разбиение строки по разделителю | 193 |
| 7.5. Замена подстрок..... | 195 |
| 7.6. Генерация всех попарных комбинаций строк..... | 195 |
| 7.7. Получение текущей даты | 197 |
| 7.8. Преобразование строки в дату | 197 |
| 7.9. Преобразование даты в строку..... | 198 |
| 7.10. Преобразование года, месяца и дня в объект Date | 199 |
| 7.11. Получение даты по юлианскому календарю | 200 |
| 7.12. Извлечение частей даты | 201 |
| 7.13. Создание последовательности дат | 202 |
| Глава 8. Вероятность | 204 |
| 8.1. Подсчет количества комбинаций..... | 206 |
| 8.2. Генерация комбинаций | 206 |
| 8.3. Генерация случайных чисел..... | 207 |
| 8.4. Генерация воспроизводимых случайных чисел | 209 |
| 8.5. Генерация случайной выборки | 210 |
| 8.6. Генерация случайных последовательностей..... | 212 |
| 8.7. Случайная перестановка вектора..... | 213 |
| 8.8. Расчет вероятностей для дискретных распределений | 213 |
| 8.9. Расчет вероятностей для непрерывных распределений..... | 215 |
| 8.10. Преобразование вероятностей в квантили | 216 |
| 8.11. Построение графика функции плотности..... | 217 |

| | |
|--|-----|
| Глава 9. Общая статистика | 222 |
| 9.1. Получение сводки данных..... | 224 |
| 9.2. Расчет относительных частот..... | 226 |
| 9.3. Представление факторов в виде таблицы и создание таблиц сопряженности | 227 |
| 9.4. Проверка категориальных переменных на независимость..... | 228 |
| 9.5. Расчет квантилей (и квартилей) набора данных | 229 |
| 9.6. Инвертирование квантиля | 230 |
| 9.7. Преобразование данных в z-оценки | 230 |
| 9.8. Проверка среднего значения выборки (t-критерий)..... | 231 |
| 9.9. Формирование доверительного интервала для среднего значения | 233 |
| 9.10. Формирование доверительного интервала для медианы..... | 234 |
| 9.11. Тестирование доли выборки | 235 |
| 9.12. Формирование доверительного интервала для доли | 236 |
| 9.13. Проверка на нормальность..... | 237 |
| 9.14. Тест последовательностей | 238 |
| 9.15. Сравнение средних значений двух выборок..... | 239 |
| 9.16. Непараметрическое сравнение местоположений двух выборок | 241 |
| 9.17. Проверка значимости корреляции | 242 |
| 9.18. Проверка групп на предмет наличия равных пропорций | 244 |
| 9.19. Парные сравнения между средними значениями групп | 245 |
| 9.20. Проверка двух выборок, чтобы определить, принадлежат ли они одному закону распределения | 246 |
| Глава 10. Графики | 248 |
| 10.1. Создание точечной диаграммы | 252 |
| 10.2. Добавление заголовка и меток..... | 253 |
| 10.3. Добавление (или удаление) координатной сетки..... | 254 |
| 10.4. Применение темы к графику ggplot | 258 |
| 10.5. Создание точечной диаграммы | 261 |
| 10.6. Добавление (или удаление) условных обозначений | 263 |
| 10.7. Построение регрессионной линии точечной диаграммы | 267 |
| 10.8. Построение точечных диаграмм..... | 270 |
| 10.9. Создание по одной точечной диаграмме | 272 |
| 10.10. Создание гистограммы | 274 |
| 10.11. Добавление доверительных интервалов в гистограмму..... | 276 |
| 10.12. Раскраска гистограммы | 279 |
| 10.13. Построение линии из точек x и y..... | 281 |
| 10.14. Изменение типа, ширины или цвета линии | 282 |
| 10.15. Построение нескольких наборов данных | 285 |
| 10.16. Добавление вертикальных или горизонтальных линий | 286 |
| 10.17. Создание диаграммы размаха | 288 |
| 10.18. Создание диаграммы размаха для каждого уровня фактора..... | 290 |
| 10.19. Создание гистограммы..... | 292 |
| 10.20. Добавление оценки плотности к гистограмме | 293 |
| 10.21. Создание графика квантиль-квантиль..... | 295 |
| 10.22. Создание других графиков квантиль-квантиль..... | 297 |
| 10.23. Построение переменной в нескольких цветах | 300 |
| 10.24. График функции | 302 |
| 10.25. Отображение нескольких графиков на одной странице | 304 |

| | |
|---|------------|
| Глава 11. Линейная регрессия и дисперсионный анализ | 309 |
| 11.1. Простая линейная регрессия | 311 |
| 11.2. Множественная линейная регрессия | 313 |
| 11.3. Получение регрессионной статистики | 314 |
| 11.4. Общая информация о регрессии | 318 |
| 11.5. Линейная регрессия без свободного члена | 321 |
| 11.6. Регрессия только тех переменных, которые сильно коррелируют с вашей зависимой переменной | 322 |
| 11.7. Линейная регрессия с эффектами взаимодействия | 325 |
| 11.8. Выбор наиболее подходящих переменных регрессии | 327 |
| 11.9. Регрессия для подмножества данных | 331 |
| 11.10. Использование выражения в формуле регрессии | 333 |
| 11.11. Полиномиальная регрессия | 334 |
| 11.12. Регрессия на преобразованных данных | 335 |
| 11.13. Поиск наиболее подходящего степенного преобразования (тест Бокса–Кокса) | 337 |
| 11.14. Формирование доверительных интервалов для коэффициентов регрессии | 342 |
| 11.15. Построение невязок регрессии | 342 |
| 11.16. Диагностика линейной регрессии | 344 |
| 11.17. Обнаружение влиятельных наблюдений | 348 |
| 11.18. Тестирование невязок на наличие автокорреляции (критерий Дарбина–Уотсона) | 349 |
| 11.19. Предсказываем новые значения | 351 |
| 11.20. Формирование интервалов предсказаний | 352 |
| 11.21. Однофакторный дисперсионный анализ | 352 |
| 11.22. Создание диаграммы взаимодействия | 354 |
| 11.23. Находим различия между средними значениями групп | 356 |
| 11.24. Устойчивый дисперсионный анализ (критерий Краскела–Уоллиса) | 358 |
| 11.25. Сравнение моделей с использованием функции <code>anova</code> | 360 |
| Глава 12. Полезные хитрости | 362 |
| 12.1. Просмотр данных | 362 |
| 12.2. Вывод на экран результата присваивания | 364 |
| 12.3. Суммирование строк и столбцов | 365 |
| 12.4. Вывод данных в столбцах | 366 |
| 12.5. Объединение данных | 367 |
| 12.6. Поиск положения определенного значения | 368 |
| 12.7. Выбор каждого n -го элемента вектора | 368 |
| 12.8. Поиск минимумов или максимумов | 369 |
| 12.9. Генерация всех комбинаций нескольких переменных | 371 |
| 12.10. Преобразование таблицы данных | 372 |
| 12.11. Сортировка таблицы данных | 373 |
| 12.12. Удаление атрибутов из переменной | 374 |
| 12.13. Раскрываем структуру объекта | 375 |
| 12.14. Определяем время выполнения кода | 378 |
| 12.15. Избавляемся от предупреждений и сообщений об ошибках | 379 |
| 12.16. Извлечение аргументов функции из списка | 380 |
| 12.17. Определение собственных бинарных операторов | 382 |
| 12.18. Избавляемся от сообщения о запуске | 383 |
| 12.19. Получение и настройка переменных среды | 384 |

| | |
|---|-----|
| 12.20. Разбиение кода на секции | 385 |
| 12.21. Локальная параллелизация выполнения кода | 386 |
| 12.22. Удалённая параллелизация выполнения кода | 388 |

Глава 13. За пределами основных цифр и статистики 392

| | |
|---|-----|
| 13.1. Минимизация или максимизация однопараметрической функции | 392 |
| 13.2. Минимизация или максимизация многопараметрической функции | 393 |
| 13.4. Метод главных компонент | 396 |
| 13.5. Простая ортогональная регрессия | 397 |
| 13.6. Поиск кластеров в данных | 399 |
| 13.7. Прогнозирование бинарной переменной (логистическая регрессия) | 402 |
| 13.8. Бутстрэппинг | 404 |
| 13.9. Факторный анализ | 406 |

Глава 14. Анализ временных рядов 411

| | |
|---|-----|
| 14.1. Представление данных временного ряда | 412 |
| 14.2. Построение данных временных рядов | 415 |
| 14.3. Извлечение самых старых или самых последних наблюдений | 417 |
| 14.4. Выбор элементов из временного ряда | 419 |
| 14.5. Объединение нескольких временных рядов | 421 |
| 14.6. Заполнение временного ряда | 423 |
| 14.7. Смещение временного ряда | 425 |
| 14.8. Вычисление последовательных различий | 427 |
| 14.9. Выполнение расчетов по временным рядам | 428 |
| 14.10. Вычисление скользящей средней | 429 |
| 14.11. Применение функции по календарному периоду | 431 |
| 14.12. Применение функции <code>rollapply</code> | 433 |
| 14.13. Построение функции автокорреляции | 434 |
| 14.14. Тестирование временного ряда на наличие автокорреляций | 436 |
| 14.15. Построение функции частичной автокорреляции | 437 |
| 14.16. Поиск корреляций с временным лагом между двумя временными рядами | 439 |
| 14.17. Удаление тренда из временного ряда | 440 |
| 14.18. Подгонка модели ARIMA | 443 |
| 14.19. Удаление незначимых коэффициентов ARIMA | 446 |
| 14.20. Выполнение диагностики для модели ARIMA | 448 |
| 14.21. Прогнозирование по модели ARIMA | 450 |
| 14.22. Построение прогноза | 451 |
| 14.23. Тестирование на наличие возвращения к среднему | 452 |
| 14.24. Сглаживание временного ряда | 455 |

Глава 15. Простое программирование 457

| | |
|--|-----|
| 15.1. Выбор из двух альтернатив: <code>if/else</code> | 458 |
| 15.2. Итерация с помощью цикла | 460 |
| 15.3. Определение функции | 461 |
| 15.4. Создание локальной переменной | 462 |
| 15.5. Выбор из нескольких альтернатив: функция <code>switch</code> | 463 |
| 15.6. Определение значений по умолчанию для параметров функции | 464 |
| 15.7. Подать сигнал с помощью сообщения об ошибке | 465 |

| | |
|--|------------|
| 15.8. Защита от ошибок | 466 |
| 15.9. Создание анонимной функции | 467 |
| 15.10. Создание коллекции многократно используемых функций | 468 |
| 15.11. Автоматическое форматирование кода | 469 |
| Глава 16. R Markdown и публикации | 471 |
| 16.1. Создание нового документа | 472 |
| 16.2. Добавление заголовка, автора или даты | 474 |
| 16.3. Форматирование текста документа | 476 |
| 16.4. Вставка заголовков документов | 476 |
| 16.5. Вставка списка | 477 |
| 16.6. Вывод результатов из кода R | 478 |
| 16.7. Контролируем, какой код и результаты отображаются | 480 |
| 16.8. Вставка графика | 481 |
| 16.9. Вставка таблицы | 484 |
| 16.10. Вставка таблицы данных | 485 |
| 16.11. Вставка математических уравнений | 488 |
| 16.12. Генерация вывода HTML | 488 |
| 16.13. Генерация вывода в формате PDF | 490 |
| 16.14. Генерация вывода в формате Microsoft Word | 492 |
| 16.15. Генерация выходных данных презентации | 498 |
| 16.16. Создание параметризованного отчета | 500 |
| 16.17. Организация рабочего процесса в R Markdown | 503 |
| Об авторах | 506 |
| Колофон | 506 |
| Предметный указатель | 507 |
| От редакции | 508 |

Предисловие редактора

Более двадцати лет назад, ещё в прошлом тысячелетии, в далёкой, далёкой, почти сказочной Новой Зеландии два статистика Джентельмен и Айхэка из Оклендского университета породили ещё один язык программирования. Немного подумали и назвали его R.

С тех пор, для того, чтобы вы могли легко и непринуждённо посчитать медиану, построить линейную модель, найти кластеры или, страшно подумать, размножить ваш набор данных для целей бутстрэпа, десятки, а возможно, и сотни тысяч квалифицированных и разных любителей статистической обработки данных положили на алтарь с буквой R самое ценное. Да, да – своё время! Пользуйтесь этой программой с уважением.

Когда мы (группа едва знакомых по интернету) чуть больше десяти лет назад начали цикл статей «Анализ данных с R» в ещё живой и бумажный журнал Linux Format, никакой печатной продукции, да и вообще хоть сколько-нибудь исчерпывающей информации на русскоязычном пространстве по этой теме просто не было. Хотя во всём остальном мире статистики R захватывал университет за университетом. Чуть позже благодаря усилиям лидера группы биолога Алексея Шипунова из этого цикла родилась первая значительная (мы честно на тот момент так думали) R-книга на русском языке «Наглядная статистика. Используем R!», выпущенная издательством «ДМК Пресс» в 2012 году и находящаяся сейчас в открытом доступе на CRAN для всех желающих.

Поэтому отрадно сейчас видеть, что мировая литература в области R-анализа уже достаточно регулярно, хоть и с задержкой, добирается до наших краёв. R – это просто. По сути, сборник рецептов является самым замечательным форматом для его практического применения. Другое дело – сам сборник рецептов – это только начало, потому что возможности R безграничны! В области обработки статистики, естественно.

Евгений Балдин, к. ф.-м. н.

Добро пожаловать в книгу рецептов R

Язык R представляет собой мощный инструмент для статистики, графики и статистического программирования. Он ежедневно используется десятками тысяч людей для проведения серьезного статистического анализа. Это бесплатная система с открытым исходным кодом, реализация которой является коллективным достижением большого числа умных, трудолюбивых людей. Для этого языка существует свыше 10 000 доступных пакетов дополнений. R является серьезным конкурентом всех коммерческих пакетов для статической обработки данных.

Но R может расстраивать, и может быть не ясно, как выполнить множество задач, даже простых. Простые задачи становятся легкими, когда вы знаете, как это сделать, но при этом понимаете, что это слово «как» может свести с ума.

Эта книга полна практических рецептов, каждый из которых решает определенную проблему. Каждый рецепт включает в себя краткое введение в решение, а затем обсуждение, целью которого является объяснить решение и дать вам представление относительно того, как оно работает. Мы знаем, что эти рецепты полезны, и знаем, что они работают, потому что сами используем их.

Диапазон рецептов обширен. Он начинается с основных задач, а затем переходит к вводу и выводу, общей статистике, графике и линейной регрессии. Любая значительная работа с R будет включать в себя большинство этих областей или их все.

Если вы новичок, то эта книга поможет вам быстрее сориентироваться. Если вы промежуточный пользователь, эта книга будет полезна, чтобы расширить ваш кругозор и покопаться в памяти («Как снова выполнить этот тест Колмогорова–Смирнова?»).

Данная книга не является учебником по языку R, хотя вы и узнаете что-то, изучая рецепты. Это не справочное руководство, но в ней содержится много полезной информации. Это не книга по программированию на R, хотя многие рецепты полезны в сценариях, написанных на этом языке.

Наконец, эта книга не является введением в статистику. Многие рецепты предполагают, что вы знакомы с базовой статистической процедурой, если таковая имеется, и просто хотите знать, как это делается в R.

Рецепты

В большинстве рецептов используется одна или две функции R для решения конкретной задачи. Важно помнить, что мы не описываем функции подробно; скорее, мы приводим достаточно описания, чтобы решить насущную проблему. Почти каждая такая функция имеет дополнительные возможности, помимо описанных здесь, а некоторые из них обладают удивительными возможностями.

Мы настоятельно рекомендуем вам прочитать страницы справки по функциям. Скорее всего, вы узнаете что-нибудь ценное.

Каждый рецепт предлагает один из способов решения конкретной задачи. Конечно, для каждой задачи существует несколько разумных решений. Когда нам было известно несколько вариантов решения, мы обычно выбирали самое простое. Для любой задачи вы, вероятно, сможете найти несколько альтернативных решений самостоятельно. Это книга рецептов, а не Библия.

В частности, R имеет буквально тысячи дополнительных пакетов, доступных для скачивания, многие из которых реализуют альтернативные алгоритмы и статистические методы. Эта книга концентрируется на основных функциях, доступных через базовый дистрибутив, в сочетании с несколькими важными пакетами, известными под общим названием *tidyverse*.

Наиболее краткое определение этого термина дает Хэдли Уикхем (<https://blog.rstudio.com/2016/09/15/tidyverse-1-0-0/>), его создатель и один из его основных разработчиков:

Tidyverse – это набор пакетов, которые работают в гармонии, потому что они имеют общие представления данных и дизайн API. Пакет *tidyverse* разработан, чтобы упростить установку и загрузку основных пакетов из *tidyverse* одной командой. Лучшее место, чтобы познакомиться со всеми пакетами в *tidyverse* и их совместимостью, – это книга R for Data Science (<https://r4ds.had.co.nz>).

ПРИМЕЧАНИЕ ОТНОСИТЕЛЬНО ТЕРМИНОЛОГИИ

Цель каждого рецепта – решить задачу и сделать это быстро. Вместо того чтобы работать в утомительной прозе, мы иногда упорядочиваем описание с помощью терминологии, которая является правильной, но не точной. В качестве неплохого примера можно привести термин *обобщенная функция*. Мы называем `print(x)` и `plot(x)` обобщенными функциями, потому что они работают для многих видов `x`, соответственно обрабатывая каждый вид. Специалист, работающий в области теории вычислительных машин и систем, вздрогнет при виде нашей терминологии, потому что, строго говоря, это не просто «функции»; это полиморфные методы с динамической диспетчеризацией. Но если бы мы тщательно расшифровывали все эти подробности, основные решения были бы погребены под техническими деталями. Поэтому мы называем их просто функциями, что, как нам кажется, более читабельно.

Еще один пример, взятый из статистики, – это сложность, связанная с семантикой проверки статистических гипотез. Использование строгого языка теории вероятностей затруднит практическое применение некоторых тестов, поэтому мы используем более разговорный язык при описании всех статистических тестов. См. введение к главе 9 для получения дополнительной информации о том, как проверка гипотез представлена в рецептах.

Наша цель состоит в том, чтобы сделать мощь языка R доступной для широкой аудитории, сделав книгу читабельной, а не формальной. Мы надеемся, что эксперты в соответствующих областях поймут, является ли наша терминология неформальной.

ПРИМЕЧАНИЯ ПО ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ И ПЛАТФОРМАМ

Базовый дистрибутив R имеет частые и запланированные выпуски, но определение языка и реализация ядра стабильны. Рецепты, приведенные в этой книге, должны работать с любым последним выпуском базового дистрибутива.

Некоторые рецепты имеют специфические для платформы соображения, и мы тщательно их отметили. Эти рецепты в основном касаются проблем, связанных с программным обеспечением, таких как установка и настройка. Насколько нам известно, все остальные рецепты будут работать на всех трех основных платформах для R: Windows, macOS и Linux/Unix.

ДОПОЛНИТЕЛЬНЫЕ ИСТОЧНИКИ

Ниже приводится ряд советов для получения дополнительной информации.

В сети

Своего рода «основа» для всего, что связано с R, – сайт проекта R (<https://www.r-project.org>). Отсюда вы можете скачать R для своей платформы, дополнительные пакеты, документацию и исходный код, а также множество других ресурсов.

Помимо сайта проекта R, мы рекомендуем использовать R-поисковик, например RSeek (<https://rseek.org>), созданный Сашей Гудманом. Вы можете использовать общую поисковую систему, такую как Google, но при применении «R» в качестве поискового запроса выдается слишком много посторонних вещей. См. рецепт 1.11 для получения дополнительной информации о поиске в интернете.

Чтение блогов – отличный способ узнать о R и быть в курсе самых передовых разработок. Таких блогов на удивление много, поэтому мы рекомендуем два из них: R-bloggers (<https://www.r-bloggers.com>), созданный Талом Галили, и PlanetR (<http://planet.r.stderr.org>). Подписавшись на их RSS-каналы, вы будете получать уведомления об интересных и полезных статьях с десятков сайтов.

Книги

Есть много, много книг об обучении и использовании R. Перечисленные здесь несколько изданий мы нашли полезными. Обратите внимание, что сайт проекта R содержит обширную библиографию книг, связанных с R (<https://www.r-project.org/doc/bib/R-books.html>).

R for Data Science (<http://shop.oreilly.com/product/0636920034407.do>) Хэдли Уикхэма и Гарретта Гроулмунда (издательство *O'Reilly*) – отличное введение в пакеты *tidyverse*, особенно в том, что касается их использования в анализе данных и статистике. Данное издание также доступно онлайн (<https://r4ds.had.co.nz>).

Мы полагаем, что книга *R Graphics Cookbook* (<http://shop.oreilly.com/product/0636920063704.do>) Уинстона Чанга (издательство *O'Reilly*) незаменима для создания графики. Книга *ggplot2: Elegant Graphics for Data Analysis* Хэдли Уикхэма (издательство *Springer*) представляет собой наиболее полный справочник по графическому пакету *ggplot2*, который мы используем в этой книге. Любому, кто занимается серьезной графикой в R, понадобится книга *R Graphics* Пола Мёррелла (издательство *Chapman & Hall / CRC*).

R in a Nutshell (<http://shop.oreilly.com/product/9780596801717.do>) Джозефа Адлера

(издательство *O'Reilly*) – это краткое руководство и справочник, который всегда будет рядом с вами. Оно охватывает гораздо больше тем, нежели эта книга рецептов. Новые книги по программированию на языке R появляются регулярно. Мы предлагаем *Hands On Programming with R* (<https://www.oreilly.com/library/view/hands-on-programming-with/9781449359089/>) Гарретта Гроулмунда (издательство *O'Reilly*) для введения или *The Art of R Programming* Нормала Мэтлоффа (издательство *No Starch Press*). *Advanced R* Хэдли Уикхэма (издательство *Chapman & Hall / CRC*) доступна в печатном виде, а также бесплатно на странице <http://adv-r.had.co.nz>. В ней еще более подробно представлены продвинутое темы R. *Efficient R Programming* (<http://shop.oreilly.com/product/0636920047995.do>) Колина Гиллеспы и Робина Ловеласа (издательство *O'Reilly*) – еще одно хорошее руководство для изучения более глубоких концепций программирования на языке R. *Modern Applied Statistics with S*, 4-е изд., Уильяма Венейблса и Брайана Рипли (Springer) использует R для иллюстрации множества передовых статистических методов. Функции и наборы данных книги доступны в пакете MASS, который входит в стандартный дистрибутив R. Серьезные фанаты могут скачать *R Language Definition* (<https://cran.r-project.org/doc/manuals/R-lang.pdf>) от R Core Team. *R Language Definition* находится в стадии разработки, но может ответить на многие ваши подробные вопросы о R как о языке программирования.

Книги по статистике

Для изучения статистики прекрасно подойдет *Using R for Introductory Statistics* Джона Верзани (издательство *Chapman & Hall / CRC*). Эта книга обучает статистике и R, давая необходимые навыки работы с компьютером для применения статистических методов.

Вам понадобится хороший учебник по статистике или справочник, чтобы точно интерпретировать статистические тесты, выполненные в R. Подобных хороших книг множество – их слишком много, чтобы можно было рекомендовать какую-либо одну.

Авторы книг по статистике все чаще используют R для иллюстрации своих методов. Если вы работаете в специализированной области, то, вероятно, найдете полезную и актуальную книгу в библиографии проекта R (<https://www.r-project.org/doc/bib/R-books.html>).

УСЛОВНЫЕ ОБОЗНАЧЕНИЯ И СОГЛАШЕНИЯ, ПРИНЯТЫЕ В КНИГЕ

В книге используются следующие типографские соглашения.

Курсив используется для смыслового выделения новых терминов, адресов электронной почты, а также имен и расширений файлов.

Моноширинный шрифт применяется для листингов программ, а также в обычном тексте для обозначения имен переменных, функций, типов, объектов, баз данных, переменных среды, операторов и ключевых слов.

Моноширинный полужирный шрифт используется для обозначения команд или фрагментов текста, которые пользователь должен ввести дословно без изменений.

Моноширинный курсив – для обозначения в исходном коде или в командах шаблонных меток-заполнителей, которые должны быть заменены соответствующими контексту реальными значениями.



Этот элемент означает подсказку или предложение.



Этот элемент означает общее примечание.



Этот элемент указывает на предупреждение или предостережение.

ИСПОЛЬЗОВАНИЕ ПРИМЕРОВ КОДА

Дополнительный материал (примеры кода, исходный код, упражнения и т. д.) доступен для скачивания на странице <http://rc2e.com>. Аккаунт в Twitter, посвященный этой книге, – @R_cookbook (https://twitter.com/R_cookbook).

Данная книга призвана помочь вам выполнить свою работу. В общем, вы можете использовать код из этой книги в своих программах и документации. Вам не нужно обращаться к нам за разрешением, если вы не воспроизводите значительную часть кода. Например, для написания программы, в которой используется несколько фрагментов кода из этой книги, оно не требуется. Продажа или распространение CD-ROM с примерами из книг O'Reilly требует разрешения. Чтобы ответить на вопрос, сославшись на эту книгу и приведя пример кода, разрешение не требуется. Включение значительного количества примеров кода из этой книги в документацию вашего продукта требует разрешения.

Атрибуция желательна, но не является обязательной. Обычно она включает в себя название книги, автора, издателя и ISBN. Например: «Книга рецептов R, 2-е изд., Дж. Д. Лонг и Пол Титор. Copyright 2019 Дж. Лонг и Пол Титор, 978-1-492-04068-2».

Если вы считаете, что применение примеров кода выходит за рамки добросовестного использования или только что описанного разрешения, свяжитесь с нами по адресу permissions@oreilly.com.

ОБУЧЕНИЕ В РЕЖИМЕ ОНЛАЙН

На протяжении почти 40 лет *O'Reilly Media* (<https://www.oreilly.com>) предоставляет технологии и бизнес-тренинги, знания и анализ, чтобы помочь компаниям добиваться успеха.

Наша уникальная сеть экспертов и новаторов делится своими знаниями и опытом через книги, статьи, конференции и нашу онлайн-платформу обучения. Платформа онлайн-обучения O'Reilly предоставляет доступ по требованию к курсам обучения в режиме реального времени, углубленным способам обучения, интерактивным средам кодирования и обширной коллекции текстов и видео от O'Reilly и свыше 200 других издательств. Для получения дополнительной информации, пожалуйста, посетите сайт <http://oreilly.com>.

КАК С НАМИ СВЯЗАТЬСЯ

Пожалуйста, направляйте комментарии и вопросы относительно этой книги издателю:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (в США или Канаде)
707-829-0515 (международный или местный)
707-829-0104 (факс)

У нас есть страница этой книги в сети, где перечислены ошибки, примеры и вся дополнительная информация. Вы можете получить доступ к этой странице по адресу http://bit.ly/RCookbook_2e.

Чтобы оставить комментарий или задать технические вопросы об этой книге, отправьте электронное письмо по адресу bookquestions@oreilly.com.

Для получения дополнительной информации о наших книгах, курсах, конференциях и новостях посетите наш сайт по адресу <http://www.oreilly.com>.

Ищите нас на Facebook: <http://facebook.com/oreilly>.

Подпишитесь на нас в Twitter: <http://twitter.com/oreillymedia>.

Смотрите нас в YouTube: <http://www.youtube.com/oreillymedia>.

БЛАГОДАРНОСТИ

Выражаем благодарность сообществу R в целом и R Core Team в частности. Их бескорыстный вклад огромен. Мир статистики извлекает огромную выгоду из их работы. Участники дискуссии сообщества R Studio очень помогли в разработке идей касательно того, как объяснить многие вещи. А сотрудники и руководство R Studio оказывали поддержку во многих отношениях. Мы в долгу перед ними за все, что они дали сообществу R.

Мы хотели бы поблагодарить технических рецензентов книги: Дэвида Кёррана, Джастина Ши и майора Дасти Тернера. Их отзывы были очень важны для улучшения качества, точности и полезности этой книги. Наши редакторы, Мелисса Поттер и Рэйчел Монаган, оказали невообразимую помощь, часто не давая нам публично демонстрировать свое невежество. Нашему редактору по производству, Кристен Браун, завидуют все технические авторы благодаря ее скорости и умению работать с Markdown и Git.

Павел хотел бы поблагодарить свою семью за поддержку и терпение при создании этой книги.

Дж. Д. Лонг хотел бы поблагодарить свою жену Мэри Бет и дочь Аду за их терпение в течение всех ранних утренних часов и выходных, которые он провел с ноутбуком, работая над этой книгой.

Глава 1

Начало работы и получение справочной информации

Эта глава закладывает основу для других глав. В ней объяснено, как скачать, установить и запустить R.

Что еще более важно, в ней также объясняется, как получить ответы на ваши вопросы. Сообщество R предоставляет большое количество документации и помощи. Вы не одиноки. Вот несколько распространенных источников справочной информации.

Локальная, установленная документация

Когда вы устанавливаете R на свой компьютер, вместе с ним устанавливается огромное количество документации. Вы можете просматривать локальную документацию (рецепт 1.7) и искать ее (рецепт 1.9). Поразительно, насколько часто ищем в интернете ответ на тот или иной вопрос только для того, чтобы обнаружить, что он уже доступен в установленной документации.

Представления задач

Представление задач (<https://cran.r-project.org/web/views/>) описывает пакеты, которые относятся к одной из областей статистической работы, такой как эконометрика, медицинская визуализация, психометрия или пространственная статистика. Каждое представление написано и поддерживается экспертом в данной области. Существует свыше 35 таких представлений, поэтому, вероятно, среди них будет один или несколько, имеющих отношение к вашей области интересов. Мы рекомендуем каждому новичку найти и прочитать хотя бы одно из представлений, чтобы познакомиться с возможностями R (рецепт 1.12).

Документация пакетов

Большинство пакетов содержат полезную документацию. Документация хранится вместе с пакетами в репозиториях пакетов, таких как CRAN (<https://cran.r-project.org>), и автоматически устанавливается на ваш компьютер при установке пакета.

Сайты с вопросами и ответами на них

На сайте, содержащем вопросы и ответы на них, каждый может задать вопрос, на который могут ответить знающие люди. Читатели голосуют за ответы, поэтому со временем будут показаны лучшие ответы. Вся эта информация помечается и архивируется для поиска. Эти сайты представляют собой нечто среднее между списком рассылки и социальной сетью; канонический пример – сайт Stack Overflow (<https://stackoverflow.com>).

Сеть

В сети много информации о R, и есть специальные инструменты для ее поиска (рецепт 1.11). Информация, содержащаяся в сети, постоянно обновляется, поэтому ищите новые, улучшенные способы организации и поиска информации о языке R.

Списки рассылки

Волонтеры щедро пожертвовали много часов, чтобы ответить на вопросы новичков, которые публикуются в списках рассылки R. Списки архивируются, поэтому вы можете искать в архивах ответы на свои вопросы (рецепт 1.13).

1.1. ЗАГРУЗКА И УСТАНОВКА R

Задача

Вам необходимо установить R на свой компьютер.

Решение

Пользователи Windows и macOS могут скачать R на сайте CRAN (Comprehensive R Archive Network). Пользователи Linux и Unix могут установить пакеты R с помощью своего инструмента управления пакетами.

Windows

1. Перейдите на страницу <http://www.r-project.org/> в своем браузере.
2. Нажмите **CRAN**. Вы увидите список зеркал сайта, упорядоченных по странам.
3. Выберите ближайший к вам сайт или самый первый указанный в верхней части списка (0-Cloud), который, как правило, хорошо работает для большинства местоположений (<https://cloud.r-project.org/>).
4. Нажмите **Download R for Windows** (Скачать R для Windows) в разделе **Download and Install R** (Скачать и установить R).
5. Нажмите **base**.
6. Нажмите на ссылку, чтобы скачать последнюю версию R (файл .exe).
7. По завершении загрузки дважды щелкните файл .exe и ответьте на привычные вопросы.

macOS

1. Перейдите на страницу <http://www.r-project.org/> в своем браузере.
2. Нажмите **CRAN**. Вы увидите список зеркал сайта, упорядоченных по странам.

3. Выберите ближайший к вам сайт или самый первый указанный в верхней части списка (O-Cloud), который, как правило, хорошо работает для большинства местоположений.
4. Нажмите **Download R for (Mac) OS X** (Скачать R для (Mac) OS X).
5. Нажмите на файл *.pkg* для последней версии R в разделе **Latest release** (Последний выпуск), чтобы скачать его.
6. По завершении загрузки дважды щелкните файл *.pkg* и ответьте на привычные вопросы.

Linux или Unix

В основных дистрибутивах Linux есть пакеты для установки R. В табл. 1-1 приведены примеры.

Таблица 1-1. Дистрибутивы Linux

| Дистрибутив | Название пакета |
|--------------------|-----------------|
| Ubuntu или Debian | r-base |
| Red Hat или Fedora | R.i386 |
| SUSE | R-base |

Используйте системный менеджер пакетов для скачивания и установки пакета. Вам понадобится пароль администратора или привилегии `sudo`; в противном случае попросите системного администратора выполнить установку.

Обсуждение

Установка R в Windows или macOS проста, потому что для этих платформ существуют готовые двоичные файлы (скомпилированные программы). Вам нужно всего лишь следовать предыдущим инструкциям. Страницы CRAN также содержат ссылки на ресурсы, связанные с установкой, например задаваемые вопросы (FAQ) и советы для особых ситуаций («Работает ли R на Windows Vista / 7/8 / Server 2008?»), что может оказаться полезным.

Наиболее подходящий способ установить R в Linux или Unix – использовать диспетчер пакетов дистрибутивов Linux для установки R в виде пакета. Дистрибутивные пакеты значительно упрощают как первоначальную установку, так и последующие обновления.

В Ubuntu или Debian используйте команду `apt-get` для загрузки и установки R. Воспользуйтесь утилитой `sudo`, чтобы получить необходимые привилегии:

```
$ sudo apt-get install r-base
```

В Red Hat или Fedora используйте `yum`:

```
$ sudo yum install R.i386
```

Большинство платформ Linux также имеют графические менеджеры пакетов, которые могут оказаться более удобными.

Помимо базовых пакетов, мы рекомендуем также установить пакеты документации. Нам нравится `r-base-html` (потому что мы любим просматривать документацию с гиперссылками), а также `r-doc-html`, который устанавливает важные руководства по R локально:


```
$ sudo apt-get install r-base-html r-doc-html
```

Некоторые репозитории Linux включают в себя готовые копии пакетов R, доступных в CRAN. Мы не используем их, потому что предпочитаем получать программное обеспечение непосредственно от самого CRAN, где обычно имеются самые свежие версии.

В редких случаях вам может понадобиться выполнить сборку R с нуля. У вас может быть неподдерживаемая версия Unix, или у вас могут быть особые соображения относительно производительности или конфигурации. Процедура сборки в Linux или Unix вполне стандартна. Скачайте архив с домашней страницы вашего зеркала CRAN; он будет называться как-нибудь вроде *R-3.5.1.tar.gz*, за исключением того, что вместо 3.5.1 будет указана последняя версия. Распакуйте архив, найдите файл с именем *INSTALL* и следуйте инструкциям.

См. также

В книге *R in a Nutshell* Джозефа Адлера (издательство *O'Reilly*) содержится более подробная информация о загрузке и установке R, включая инструкции по сборке версий для Windows и macOS. Возможно, самым полным является руководство по установке и администрированию R, доступное в CRAN, где описывается сборка и установка R на различных платформах.

Это рецепт установки базового пакета. См. рецепт 3.10 для установки пакетов дополнений из CRAN.

1.2. УСТАНОВКА RSTUDIO

Задача

Вам нужна более комплексная интегрированная среда разработки (IDE), чем та, что есть в R по умолчанию. Другими словами, вы хотите установить RStudio Desktop.

Решение

За последние несколько лет RStudio стал наиболее широко используемой интегрированной средой разработки для R. Мы считаем, что почти вся работа с R должна выполняться в среде разработки RStudio Desktop, если нет веской причины поступить иначе. RStudio производит ряд продуктов, включая RStudio Desktop, RStudio Server и RStudio Shiny Server среди прочих. В этой книге мы будем использовать термин RStudio для обозначения RStudio Desktop, хотя большинство концепций применимо и к RStudio Server.

Чтобы установить RStudio, загрузите последнюю версию установщика для вашей платформы с сайта RStudio (<https://rstudio.com/products/rstudio/download/>).

Версию RStudio Desktop с открытым исходным кодом можно скачать и использовать бесплатно.

Обсуждение

В этой книге используются RStudio версии 1.2.x и R версии 3.5.x. Новые версии RStudio выпускаются каждые несколько месяцев, поэтому регулярно обновляй-

те их. Обратите внимание, что RStudio работает с любой версией R, которую вы установили, поэтому обновление до последней версии RStudio *не* обновляет вашу версию R. R необходимо обновлять отдельно.

Взаимодействие с R в RStudio немного отличается от встроенного пользовательского интерфейса R. Для этой книги мы решили использовать RStudio для всех примеров.

1.3. ЗАПУСК RSTUDIO

Задача

Вы хотите запустить RStudio на своем компьютере.

Решение

Распространенная ошибка, которую допускают новые пользователи R и RStudio, – это случайный запуск R, когда они намеревались запустить RStudio. Самый простой способ убедиться в том, что вы действительно запускаете RStudio, – это найти RStudio на своем рабочем столе, а затем использовать любой метод, который предоставляет ваша ОС для закрепления иконки там, где ее легко можно будет найти позже:

Windows

Нажмите на меню **Пуск** в левом нижнем углу экрана. В поле поиска введите RStudio.

macOs

Найдите в панели запуска приложение RStudio или нажмите **Cmd-space** (**Cmd** – это команда или клавиша **⌘**) и введите RStudio, чтобы выполнить поиск с помощью Spotlight Search.

Ubuntu

Нажмите сочетание клавиш **Alt-F1** и введите RStudio для поиска RStudio.

Обсуждение

На рис. 1-1, видно, что иконки R и RStudio похожи, что может привести вас в замешательство



Рис. 1-1. Иконки R и RStudio в macOS

Если вы нажмете на значок R, то увидите нечто похожее на то, что изображено на рис. 1-2. Это базовый интерфейс R в Mac, но, конечно же, не RStudio.

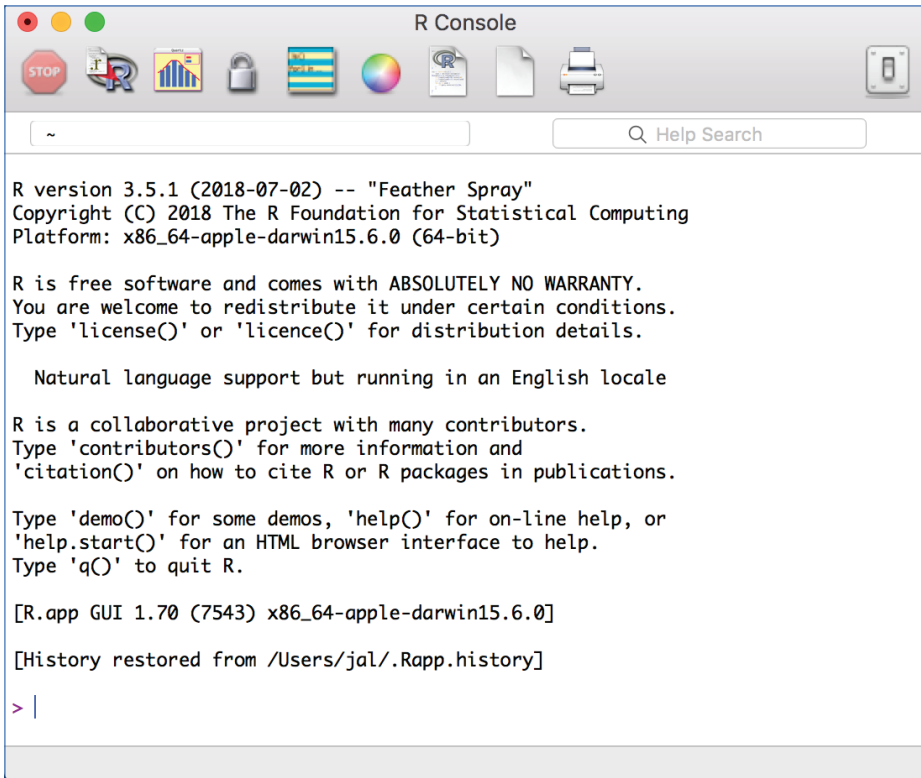


Рис. 1-2. Консоль R в macOS

Когда вы запускаете RStudio, по умолчанию он снова откроет последний проект, над которым вы работали в RStudio.

1.4. Ввод команд

Задача

Вы запустили RStudio. Что теперь?

Решение

Когда вы запускаете RStudio, главное окно в левой части экрана – это сеанс R. Оттуда вы можете вводить команды в интерактивном режиме непосредственно в R.

Обсуждение

Приглашение командной строки в R выглядит так: `>`. Для начала просто воспринимайте R как большой калькулятор: введите выражение, а R вычислит его и выведет результат:

```
> 1+1
[1] 2
>
```

Компьютер складывает 1 и 1 и отображает результат, 2.

[1] перед 2 может сбивать с толку. Для R результат – это вектор, хотя у него имеется только один элемент. R помечает значение с помощью [1], чтобы показать, что это первый элемент вектора... что неудивительно, поскольку это *единственный* элемент вектора.

R будет запрашивать ввод, пока вы не введете полное выражение. Выражение `max(1,3,5)` является полным, поэтому R прекращает чтение ввода и вычисляет то, что получил:

```
> max(1, 3,
+5)
[1] 5
>
```

`max (1,3,` напротив, представляет собой неполное выражение, поэтому R запрашивает дополнительный ввод. Вместо знака «больше, чем» (`>`) появится «плюс» (`+`), сообщая, что R ожидает большего:

```
> max(1, 3, +5)
[1] 5
>
```

Команды легко набирать неправильно, а вводить их снова утомительно и неприятно. Таким образом, R включает редактирование командной строки, чтобы сделать жизнь проще. Он определяет одиночные нажатия клавиш, которые позволяют легко вызывать, исправлять и повторно выполнять ваши команды. Типичное взаимодействие с командной строкой выглядит так:

1. Вы вводите выражение с опечаткой.
2. R жалуется на вашу ошибку.
3. Вы нажимаете клавишу со стрелкой вверх, чтобы вызвать на экран строку, в которой есть ошибка.
4. Вы используете клавиши со стрелками влево и вправо, чтобы переместить курсор назад к ошибке.
5. Вы используете клавишу **Delete**, чтобы удалить неправильные символы.
6. Вы вводите исправленные символы, которые вставляются в командную строку.
7. Вы нажимаете **Enter**, чтобы повторно выполнить исправленную команду.

Это лишь основы. R поддерживает обычные нажатия клавиш для вызова и редактирования командных строк, как показано в табл. 1-2.

В большинстве операционных систем также можно использовать мышь, чтобы выделить команды, а затем использовать обычные команды копирования и вставки, чтобы вставить текст в новую командную строку.

Таблица 1-2. Оперативные клавиши выбора в R

| Маркированная клавиша | Сочетание клавиш | Результат |
|-----------------------|------------------|---|
| Up arrow | Ctrl-P | Вспомнить предыдущую команду, перемещаясь назад по истории команд |
| Down arrow | Ctrl-N | Движение вперед по истории команд |
| Backspace | Ctrl-H | Удалить символ слева от курсора |
| Delete (Del) | Ctrl-D | Удалить символ справа от курсора |
| Home | Ctrl-A | Переместить курсор в начало строки |
| End | Ctrl-E | Переместить курсор в конец строки |
| Right arrow | Ctrl-F | Переместить курсор вправо (вперед) на один символ |
| Left arrow | Ctrl-B | Переместить курсор влево (назад) на один символ |
| | Ctrl-K | Удалить все начиная от положения курсора до конца строки |
| | Ctrl-U | Очистить всю чертову строку и начать все сначала |
| Tab | | Автодополнение имени (на некоторых платформах) |

См. также

См. рецепт 2.12. В главном меню Windows выберите **Справка** → **Консоль** для получения полного списка нажатий клавиш, полезных для редактирования в командной строке.

1.5. Выход из RStudio

Задача

Вам нужно выйти из RStudio.

Решение

Windows и большинство дистрибутивов Linux

Выберите **Select File** → **Quit Session** (Файл → Выйти из сеанса) в главном меню или нажмите **X** в верхнем правом углу окна.

macOS

Выберите **Select File** → **Quit Session** (Файл → Выйти из сеанса) в главном меню, или нажмите **Cmd-Q**, либо щелкните красный кружок в верхнем левом углу окна.

На всех платформах вы также можете использовать функцию `q` (как в `quit`) для завершения сеанса R и RStudio:

```
q()
```

Обратите внимание на пустые скобки, которые необходимы для вызова функции.

Обсуждение

Каждый раз, когда вы выходите, R обычно спрашивает, хотите ли вы сохранить рабочее пространство. У вас есть три варианта:

- сохранить рабочее пространство и выйти;
- не сохранять свое рабочее пространство и все равно выйти;
- выполнить отмену, вернувшись в командную строку, вместо того чтобы выйти.

Если вы сохраняете свое рабочее пространство, R записывает его в файл с именем `.RData` в текущем рабочем каталоге. При сохранении рабочей области сохраняются любые объекты R, которые вы создали. В следующий раз, когда вы запустите R в том же каталоге, рабочая область будет загружена автоматически. При сохранении вашей рабочей области ранее сохраненная рабочая область будет перезаписана, если таковая имеется, поэтому не сохраняйте ее, если вам не нравятся ваши изменения (например, если вы случайно удалили важные данные из своего рабочего пространства).

Мы рекомендуем никогда не сохранять рабочее пространство при выходе и вместо этого всегда в явном виде сохранять проект, сценарии и данные. Мы также рекомендуем отключить приглашение к вводу команды, чтобы сохранить и автоматически восстановить рабочее пространство в RStudio, используя глобальные параметры, которые находятся в меню **Tools** → **Global Options** (Инструменты → Глобальные параметры) и показаны на рис. 1-3. Таким образом, когда вы выходите из R и RStudio, вам не будет предложено сохранить рабочее пространство. Но имейте в виду, что любые объекты, созданные, но не сохраненные на диск, будут потеряны!

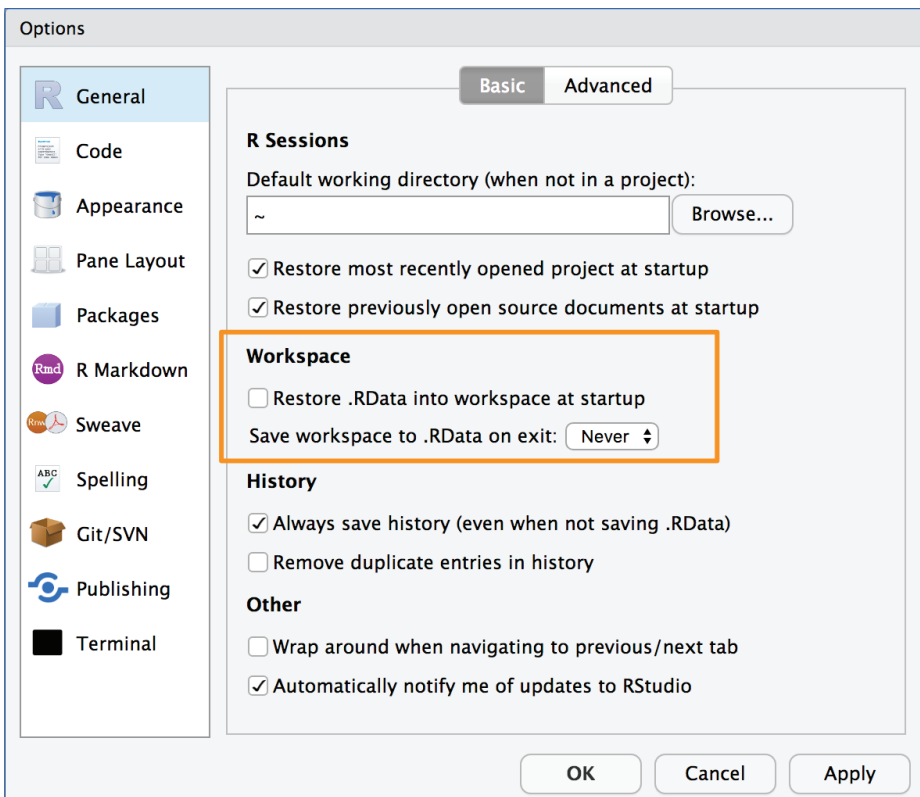


Рис. 1-3. Параметры сохранения рабочего пространства

См. также

См. рецепт 3.1 для получения дополнительной информации о текущем рабочем каталоге и рецепт 3.3 для получения дополнительной информации о сохранении рабочего пространства. Также см. вторую главу книги *R in a Nutshell*.

1.6. ПЕРЫВАНИЕ R

Задача

Вам нужно прервать длительные вычисления и вернуться в командную строку, не выходя из RStudio.

Решение

Нажмите клавишу **Esc** на клавиатуре или щелкните на меню **Session** (Сеанс) в RStudio и выберите **Interrupt R** (Прервать R). Вы также можете щелкнуть значок знака остановки в окне консоли кода.

Обсуждение

Прерывание R означает указание R прекратить выполнение текущей команды, но без удаления переменных из памяти или полного закрытия RStudio. Тем не менее прерывание R может оставить ваши переменные в неопределенном состоянии, в зависимости от того, насколько далеко продвинулись вычисления, поэтому проверьте свое рабочее пространство после прерывания.

См. также

См. рецепт 1.5.

1.7. ПРОСМОТР ПРИЛАГАЕМОЙ ДОКУМЕНТАЦИИ

Задача

Вы хотите прочитать документацию, поставляемую с R.

Решение

Используйте функцию `help.start`, чтобы увидеть оглавление документации:

```
help.start()
```

Отсюда доступны ссылки для всей установленной документации. В RStudio справка будет отображаться на панели справки, которая по умолчанию находится в правой части экрана.

В RStudio также можно нажать **Help** → **R Help** (Справка → Справка R), чтобы получить список с опциями справки как для R, так и для RStudio.

Обсуждение

Базовый дистрибутив R включает в себя множество документации – буквально тысячи страниц. При установке дополнительных пакетов эти пакеты содержат документацию, которая также установлена на вашем компьютере.

Эту документацию легко просмотреть с помощью функции `help.start`, которая открывает оглавление верхнего уровня. На рис. 1-4 показано, как `help.start` отображается на панели справки в RStudio.

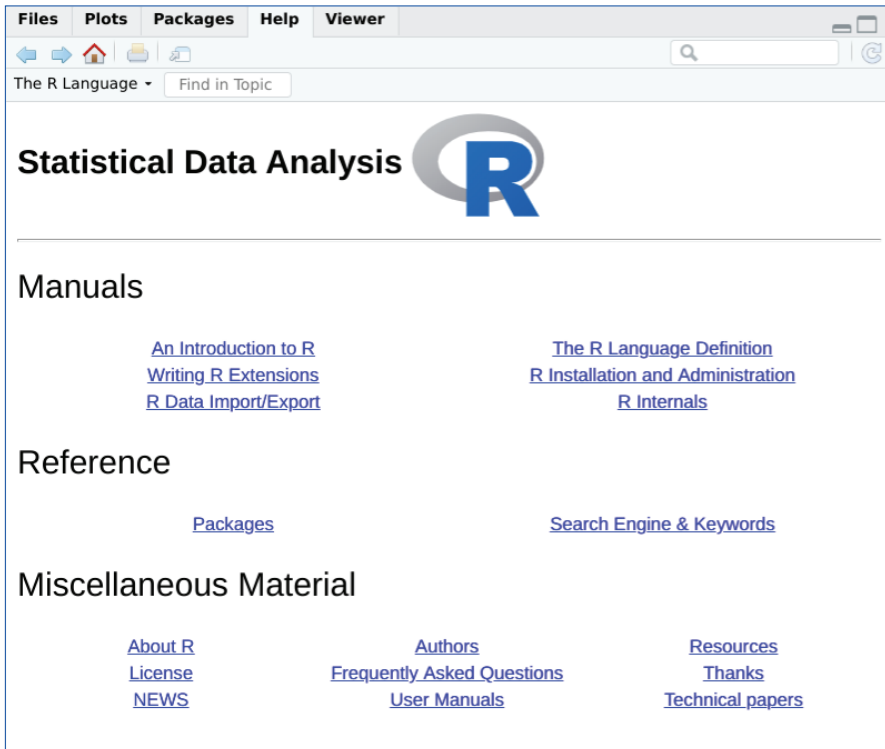


Рис. 1-4. `help.start` в RStudio

Две ссылки в разделе «Базовый справочник R» особенно полезны:

Пакеты (Packages)

Нажмите здесь, чтобы увидеть список всех установленных пакетов – как базовых, так и установленных дополнительно. Нажмите на имя пакета, чтобы увидеть список его функций и наборов данных.

Поисковая система и ключевые слова (Search Engine & Keywords)

Нажмите здесь, чтобы получить доступ к простой поисковой системе, которая позволяет выполнять поиск в документации по ключевому слову или фразе. Существует также список распространенных ключевых слов, организованных по темам; щелкните на одно из них, чтобы увидеть связанные страницы.

Базовая документация R, доступ к которой осуществляется через `help.start`, загружается на ваш компьютер при установке R. Справка RStudio, доступ к которой вы получаете с помощью опции меню **Help** → **R Help** (Справка → Справка R), представляет страницу с ссылками на сайт RStudio. Таким образом, вам понадобится доступ в интернет для доступа к справочным ссылкам RStudio.

См. также

Локальная документация скопирована с сайта R Project (<https://www.r-project.org>), где могут быть обновленные документы.

1.8. ПОЛУЧЕНИЕ СПРАВКИ ПО ФУНКЦИИ

Задача

Вы хотите больше узнать о функции, которая установлена на вашем компьютере.

Решение

Используйте функцию `help` для отображения документации по функции:

```
help(имяфункции)
```

Используйте функцию `args` для быстрого напоминания аргументов функции:

```
args(имяфункции)
```

Используйте функцию `example`, чтобы увидеть примеры использования функции:

```
example(имяфункции)
```

Обсуждение

В этой книге мы представляем много функций R. Каждая функция имеет больше дополнительных свойств, чем мы можем описать. Если функция заинтересовала вас, мы настоятельно рекомендуем прочитать страницу справки для этой функции. Одно из таких свойств может оказаться очень полезным для вас.

Предположим, вы хотите узнать больше о функции `mean`. Используйте функцию `help`:

```
help mean
```

Откроется страница справки для функции `mean` на панели справки в RStudio. Для быстрого вызова команды `help` просто наберите `?`, за которым следует имя функции:

```
?mean
```

Иногда вам просто нужно быстрое напоминание об аргументах функции: каковы они и в каком порядке встречаются? Для этого случая используйте функцию `args`:

```
args(mean)
#> function (x, ...)
#> NULL

args(sd)
#> function (x, na.rm = FALSE)
#> NULL
```

Первая строка вывода представляет собой краткий обзор вызова функции. В случае с `mean` в синопсисе показан один аргумент, `x`, который представляет собой вектор чисел. В случае с `sd` синопсис показывает тот же вектор, `x` и необязательный аргумент `na.rm`. (Вторую строку вывода, которая часто представляет собой просто `NULL`,

можно игнорировать.) В RStudio вы будете видеть вывод args в виде всплывающей подсказки над курсором при вводе имени функции, как показано на рис. 1-5.

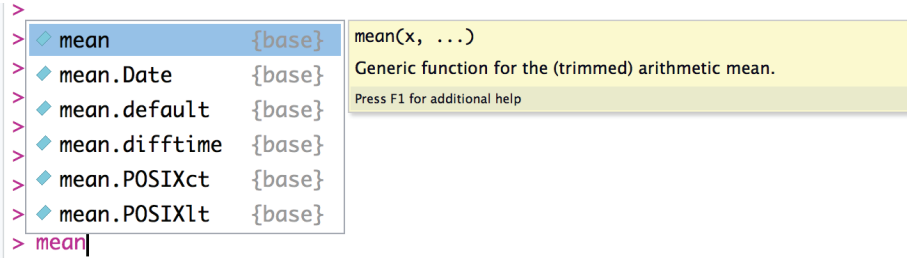


Рис. 1-5. Подсказка RStudio

Большая часть документации по функциям содержит пример кода в конце документа. Отличная особенность R заключается в том, что вы можете запросить выполнение примеров, в результате чего увидите небольшую демонстрацию возможностей функции. Например, в документации функции `mean` содержатся примеры, но вам не нужно вводить их самостоятельно: просто используйте функцию `example`, чтобы увидеть их выполнение:

```
example(mean)
#>
#> mean> x <- c(0:10, 50)
#>
#> mean> xm <- mean(x)
#>
#> mean> c(xm, mean(x, trim = 0.10))
#> [1] 8.75 5.50
```

Все, что вы видите после `example(mean)`, было создано R, который выполнил примеры со страницы справки и отобразил результаты.

См. также

См. рецепт 1.9 для поиска функций и рецепт 3.6 для получения дополнительной информации о пути поиска.

1.9. ПОИСК В ПРИЛАГАЕМОЙ ДОКУМЕНТАЦИИ

Задача

Вы хотите узнать больше о функции, которая установлена на вашем компьютере, но функция `help` сообщает, что не может найти подходящую документацию.

Или же вы хотите найти в установленной документации ключевое слово.

Решение

Используйте функцию `help.search` для поиска в документации R на своем компьютере:

```
help.search("шаблон")
```

Типичным шаблоном является имя функции или ключевое слово. Обратите внимание, что он должен быть заключен в кавычки.

Для своего удобства вы также можете вызвать поиск, используя два вопросительных знака (в этом случае кавычки не требуются). Обратите внимание, что для поиска функции по имени используется один знак вопроса, а для поиска текстового шаблона – два:

```
> ??шаблон
```

Обсуждение

Время от времени вы можете запрашивать справку по функции, только чтобы получить сообщение, в котором говорится, что R ничего о ней неизвестно:

```
help(adf.test)
#> No documentation for 'adf.test' in specified packages and libraries:
#> you could try '??adf.test'
```

Если вы знаете, что функция установлена на вашем компьютере, это может быть неприятно. Проблема здесь состоит в том, что пакет функции в данный момент не загружен, и вы не знаете, в каком пакете содержится эта функция. Это своего рода уловка-22 (сообщение об ошибке указывает на то, что пакет в данный момент отсутствует в вашем пути поиска, поэтому R не может найти файл справки; см. рецепт 3.6 для получения более подробной информации).

Решение заключается в поиске всех установленных пакетов для этой функции. Просто используйте функцию `help.search`, как указано в сообщении об ошибке:

```
help.search("adf.test")
```

Поиск выдаст список всех пакетов, которые содержат эту функцию:

```
Help files with alias or concept or title matching 'adf.test' using
regular expression matching:

tseries::adf.test    Augmented Dickey-Fuller Test
Type '?PKG::FOO' to inspect entry 'PKG::FOO TITLE'.
```

Предыдущий вывод показывает, что пакет `tseries` содержит функцию `adf.test`. Можно увидеть ее документацию, явно указав справке, какой пакет содержит функцию:

```
help(adf.test, package = "tseries")
```

или можно использовать оператор двойного двоеточия, чтобы указать R выполнить поиск в определенном пакете:

```
?tseries::adf.test
```

Вы можете расширить свой поиск, используя ключевые слова. Затем R найдет любую установленную документацию, которая содержит ключевые слова. Предположим, вы хотите найти все функции, в которых упоминается расширенный тест Дики–Фуллера (ADF). Можно искать по вероятному шаблону:

```
help.search("dickey-fuller")
```

См. также

Вы также можете получить доступ к локальной поисковой системе через браузер документации; см. рецепт 1.7, чтобы узнать, как это сделать. См. рецепт 3.6 для получения дополнительной информации о пути поиска и рецепт 1.8 для получения справки по функциям.

1.10. ПОЛУЧЕНИЕ СПРАВКИ ПО ПАКЕТУ

Задача

Вы хотите узнать больше о пакете, установленном на вашем компьютере.

Решение

Используйте функцию `help` и укажите имя пакета (без имени функции):

```
help(package = "имяпакета")
```

Обсуждение

Иногда вам нужно знать содержимое пакета (функции и наборы данных).

Это особенно верно, например, после загрузки и установки нового пакета. Функция `help` может предоставить содержимое плюс другую информацию, как только вы укажете имя пакета.

В результате этого вызова функции `help` будет показана информация для пакета `tseries`, стандартного пакета в базовом дистрибутиве:

```
help(package = "tseries")
```

Эта информация начинается с описания и продолжается указателем функций и наборов данных. В RStudio страница справки в формате HTML откроется в окне справки интегрированной среды разработки.

Некоторые пакеты также содержат сопроводительную документацию, это введения, учебные пособия или справочные карточки. Они устанавливаются на ваш компьютер как часть документации пакета при его инсталляции. Страница справки для пакета содержит список сопроводительной документации в нижней части.

Вы можете просмотреть этот список на своем компьютере, используя функцию `vignette`:

```
vignette()
```

В RStudio откроется новая вкладка со списком всех пакетов, установленных на вашем компьютере, которые содержат виньетки, а также их названия и описания.

Можно увидеть документацию для конкретного пакета, включив его название:

```
vignette(package = "имяпакета")
```

У каждой виньетки есть имя, которое используется для ее просмотра:

```
vignette("имявиньетки")
```

См. также

См. рецепт 1.8 для получения справки по конкретной функции в пакете.

1.11. ПОИСК СПРАВКИ В ИНТЕРНЕТЕ

Задача

Вам нужно найти в интернете информацию и ответы, касающиеся R.

Решение

Внутри R используйте функцию `RSiteSearch` для поиска по ключевому слову или фразе:

```
RSiteSearch("ключевая фраза")
```

В своем браузере попробуйте использовать приведенные ниже сайты для поиска:

RSeek (<https://rseek.org>)

Это пользовательская поисковая система Google, ориентированная на сайты, посвященные R.

Stack Overflow (<https://stackoverflow.com>)

Stack Overflow – сайт с вопросами и ответами с возможностью поиска от сети Stack Exchange, который ориентирован на такие проблемы программирования, как структуры данных, кодирование и графика. Stack Overflow – отличная «первая остановка» для всех ваших вопросов, касающихся синтаксиса.

Cross Validated (<https://stats.stackexchange.com>)

Cross Validated – это сайт Stack Exchange, нацеленный на статистику, машинное обучение и анализ данных, а не на программирование. Это хорошее место, чтобы задать вопрос по поводу того, какой статистический метод использовать.

Сообщество RStudio (<https://community.rstudio.com>)

Сайт сообщества RStudio – это дискуссионный форум, организованный RStudio. Темы включают в себя R, RStudio и связанные с ними технологии. Поскольку этот форум посвящен RStudio, его часто посещают сотрудники компании RStudio Inc и те, кто нередко использует это программное обеспечение. Это неплохое место для вопросов общего характера и вопросов, которые, возможно, не вписываются в формат, ориентированный на синтаксис Stack Overflow.

Обсуждение

Функция `RSiteSearch` откроет окно браузера и направит его в поисковую систему на сайте R Project (<http://search.r-project.org>). Здесь вы можете уточнить параметры поиска. Например, так мы ищем термин «каноническая корреляция»:

```
RSiteSearch("canonical correlation")
```

Это очень удобно. Таким образом можно выполнять быстрый поиск, не выходя из R. Однако область поиска ограничена документацией R и архивами списков рассылки.

RSeek (<https://rseek.org>) обеспечивает более широкий поиск. Его достоинство заключается в том, что он использует всю мощь поисковой системы Google, концентрируясь на сайтах, имеющих отношение к R, и исключает посторонние результаты общего поиска Google. Прелесть RSeek в том, что он организует результаты полезным способом.

На рис. 1-6 показаны результаты посещения сайта RSeek и поиска термина «корреляция». Обратите внимание, что вкладки в верхней части позволяют переходить к различным типам контента:

- All results (Все результаты);
- Packages (Пакеты);
- Books (Книги);
- Support (Служба поддержки);
- Articles (Статьи);
- For Beginners (Для начинающих).

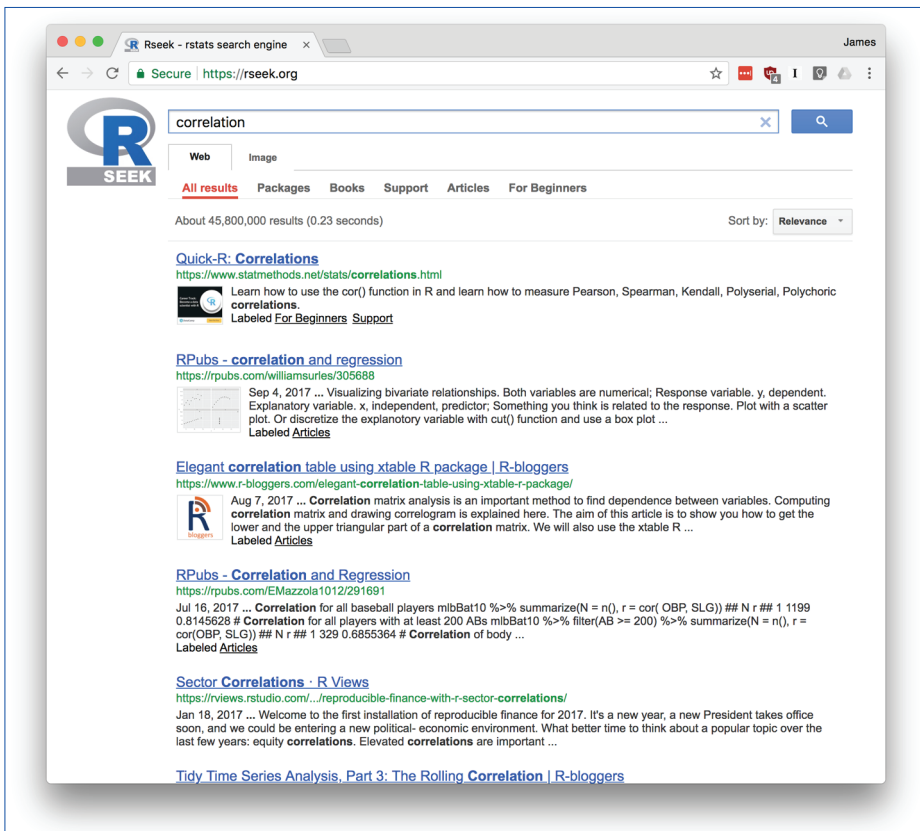


Рис. 1-6. RSeek

Stack Overflow (<https://stackoverflow.com>) – это сайт с вопросами и ответами. Это означает, что любой может отправить вопрос, а опытные пользователи дадут на него ответы – часто на каждый вопрос есть несколько ответов. Читатели голосят за ответы, поэтому хорошие ответы обычно поднимаются наверх, что созда-

ет богатую базу данных диалогов, состоящих из вопросов и ответов, по которым можно выполнять поиск. Stack Overflow ориентирован главным образом на решение проблем, и темы прежде всего касаются программирования на языке R.

Stack Overflow содержит вопросы по многим языкам программирования; поэтому при вводе термина в поле поиска добавьте к нему «[r]», чтобы сфокусировать поиск на вопросах, касающихся R. Например, при поиске «стандартная ошибка [r]» будут выбраны только вопросы, помеченные как касающиеся языка R, и вам не будут попадаться вопросы по Python и C++.

Stack Overflow также включает в себя статью о языке R (<https://stackoverflow.com/tags/r/info>), которая предоставляет превосходный список онлайн-ресурсов для сообщества.

У Stack Exchange (материнская компания Stack Overflow) есть сайт для статистического анализа под названием Cross Validated (<https://stats.stackexchange.com>). Этот сайт больше сфокусирован на статистике, нежели на программировании, поэтому используйте его при поиске ответов, которые больше касаются статистики в целом, а не R в частности.

У RStudio также есть собственный форум (<https://community.rstudio.com>). Это отличное место, чтобы задавать общие и более концептуальные вопросы, которые могут не дать нужного результата на Stack Overflow.

См. также

Если в результате поиска вы обнаружите полезный пакет, используйте рецепт 3.10, чтобы установить его на свой компьютер.

1.12. ПОИСК СООТВЕТСТВУЮЩИХ ФУНКЦИЙ И ПАКЕТОВ

Задача

Из 10 000+ пакетов для R вы не знаете, какие из них будут вам полезны.

Решение

Чтобы найти пакеты, относящиеся к определенной области, посетите список представлений задач CRAN (<https://cran.r-project.org/web/views/>). Выберите представление задачи для вашей области, которая даст вам описания соответствующих пакетов и ссылки на них. Или посетите сайт RSeek (<https://rseek.org>), выполните поиск по ключевому слову, перейдите на вкладку **Представления задач** и выберите подходящее представление.

Посетите сайт <https://crantastic.org> и поищите пакеты по ключевому слову.

- Чтобы найти соответствующие функции, посетите сайт RSeek, выполните поиск по имени или ключевому слову и нажмите на вкладку **Функции**.

Обсуждение

Эта проблема особенно неприятна для начинающих. Вы думаете, что R может решить ваши проблемы, но не знаете, какие пакеты и функции были бы полезны. Распространенный вопрос, встречающийся в списках рассылки: «Существует ли какой-нибудь пакет для решения проблемы X?» Это тихий крик того, кто тонет в R.

На момент написания этих строк доступно свыше 10 000 пакетов для бесплатной загрузки с CRAN. У каждого пакета имеется страница с кратким описанием и ссылками на документацию. Найдя потенциально интересный пакет, вы обычно нажимаете ссылку **Reference manual** (Справочное руководство), чтобы просмотреть документацию в формате PDF со всеми подробностями. (Сводная страница также содержит ссылки на скачивание для установки пакета, но вы редко устанавливаете пакет таким образом; см. рецепт 3.10.)

Иногда у вас просто есть общий интерес – например, байесовский анализ, эконометрика, оптимизация или графика. CRAN содержит набор страниц с представлениями задач, описывающих пакеты, которые могут быть полезны. Представление задач – отличное место для начала, поскольку вы получаете обзор того, что доступно. Вы можете просмотреть список страниц с представлениями задач по адресу <https://cran.r-project.org/web/views/> или выполнить их поиск, как описано в решении. Представления задач CRAN перечисляют ряд широких сфер деятельности и показывают пакеты, которые используются в каждой области. Например, существуют представления задач для высокопроизводительных вычислений, генетики, временных рядов и социальных наук, и это лишь некоторые из них.

Предположим, вы знаете название полезного пакета, скажем, увидев его в интернете. Полный список пакетов в алфавитном порядке доступен по адресу <https://cran.r-project.org/web/packages/>.

См. также

Вы можете скачать и установить пакет R под названием `sos`, который предоставляет другие мощные способы поиска пакетов; см. <https://cran.r-project.org/web/packages/sos/vignettes/sos.pdf>.

1.13. ПОИСК В СПИСКАХ РАССЫЛКИ

Задача

У вас есть вопрос, и вы хотите найти его в архивах списков рассылки, чтобы узнать, был ли ранее дан ответ на ваш вопрос.

Решение

Перейдите на страницу <https://r.789695.n4.nabble.com>. Выполните поиск по ключевому слову или другому поисковому запросу из вашего вопроса. Вы увидите результаты из списков рассылки поддержки.

Обсуждение

Данный рецепт в действительности является всего лишь приложением рецепта 1.11. Но это важное приложение, потому что вы должны выполнить поиск в архивах списка рассылки, прежде чем отправлять новый вопрос в список. На ваш вопрос, вероятно, уже есть ответ.

См. также

В CRAN есть список дополнительных ресурсов для поиска в сети; см. <https://cran.r-project.org/search.html>.

1.14. ОТПРАВКА ВОПРОСОВ В STACK OVERFLOW ИЛИ В ДРУГОЕ МЕСТО В СООБЩЕСТВЕ

Задача

У вас есть вопрос, на который вы не можете найти ответ в интернете, поэтому вы хотите отправить вопрос сообществу R.

Решение

Первое, что необходимо сделать, чтобы задать вопрос онлайн, – создать воспроизводимый пример. Наличие примера кода, который можно выполнить и четко увидеть вашу проблему, является наиболее важной частью обращения за помощью онлайн. Вопрос с хорошим воспроизводимым примером состоит из трех компонентов:

Пример данных

Это могут быть смоделированные данные или какие-либо реальные данные, которые вы предоставляете.

Пример кода

Этот код показывает, что вы пытались сделать что-то, или ошибку, которую вы получаете.

Письменное описание

Здесь вы объясняете, что у вас есть, что бы вы хотели и что вы пробовали сделать, но это не работало.

Подробности написания воспроизводимого примера приведены в разделе «Обсуждение». Получив воспроизводимый пример, вы можете опубликовать свой вопрос на сайте Stack Overflow (https://stackoverflow.com/users/login?src=anon_ask&returnurl=https%3a%2f%2fstackoverflow.com%2fquestions%2fask). Обязательно включите тег `r` в разделе «Теги» на странице вопросов.

Если ваш вопрос носит более общий характер или связан с концепциями, а не с конкретным синтаксисом, RStudio организует дискуссионный форум сообщества RStudio (<https://community.rstudio.com>). Обратите внимание, что этот сайт разбит на несколько тем, поэтому выберите категорию тем, которая наилучшим образом соответствует вашему вопросу.

Вы также можете отправить свой вопрос в списки рассылки R (но не отправляйте его на несколько сайтов, в списки рассылки и Stack Overflow, поскольку это воспринимается как грубый кросспостинг).

Страница списков рассылки (<https://www.r-project.org/mail.html>) содержит общую информацию и инструкции по использованию списка рассылки Rhelp. Вот как выглядит этот процесс в общих чертах:

1. Подпишитесь на основной список рассылки R, R-help (<https://stat.ethz.ch/mailman/listinfo/r-help>).
2. Тщательно и правильно напишите свой вопрос и включите туда свой воспроизводимый пример.
3. Отправьте свой вопрос по адресу `r-help@r-project.org`.

Обсуждение

Список рассылки R-help, Stack Overflow и сайт сообщества RStudio – отличные ресурсы, но, пожалуйста, относитесь к ним как к последнему средству. Прочтите страницы справки, документацию, выполните поиск в архивах и в интернете. Скорее всего, на ваш вопрос уже есть ответ. Не обманывайте себя: очень немногие вопросы уникальны. Если вы исчерпали все другие варианты, возможно, пришло время создать хороший вопрос.

Воспроизводимый пример – основа хорошего запроса о помощи. Первый компонент – пример данных. Хороший способ получить его – смоделировать данные, используя несколько функций R.

В приведенном ниже примере мы создали таблицу данных `example_df` с тремя столбцами, каждый из которых имеет свой тип данных:

```
set.seed(42)
n <- 4
example_df <- data.frame(
  some_reals = rnorm(n),
  some_letters = sample(LETTERS, n, replace = TRUE),
  some_ints = sample(1:10, n, replace = TRUE)
)
example_df
#>   some_reals some_letters some_ints
#> 1   1.371         R           10
#> 2  -0.565         S            3
#> 3   0.363         L            5
#> 4   0.633         S           10
```

Обратите внимание, что в этом примере вначале используется команда `set.seed`. Это гарантирует, что при каждом запуске этого кода ответы будут одинаковыми. Значение `n` – это число строк примеров данных, которые вы хотели бы создать. Сделайте ваш пример данных максимально простым, чтобы проиллюстрировать свой вопрос.

Альтернативой созданию смоделированных данных является использование примера данных, поставляемых с R. Например, набор данных `mtcars` содержит таблицу данных с 32 записями о различных моделях автомобилей:

```
data(mtcars)
head(mtcars)
#>      mpg   cyl  disp    hp  drat   wt    qsec    vs  am  gear carb
#> Mazda RX4      21.0   6  160   110 3.90  2.62  16.5   0   1   4     4
#> Mazda RX4 Wag  21.0   6  160   110 3.90  2.88  17.0   0   1   4     4
#> Datsun 710     22.8   4  108   93  3.85  2.32  18.6   1   1   4     1
#> Hornet 4 Drive  21.4   6  258  110 3.08  3.21  19.4   1   0   3     1
#> Hornet Sportabout 18.7   8  360  175 3.15  3.44  17.0   0   0   3     2
#> Valiant       18.1   6  225  105 2.76  3.46  20.2   1   0   3     1
```

Если ваш пример воспроизводим только с вашими собственными данными, вы можете использовать `dput`, чтобы поместить часть собственных данных в строку, которую можно использовать в своем примере. Мы проиллюстрируем этот подход, используя две строки из набора данных `mtcars`:

```
dput(head(mtcars, 2))
#> structure(list(mpg = c(21, 21), cyl = c(6, 6), disp = c(160,
```

```
#> 160), hp = c(110, 110), drat = c(3.9, 3.9), wt = c(2.62, 2.875
#> ), qsec = c(16.46, 17.02), vs = c(0, 0), am = c(1, 1), gear = c(4,
#> 4), carb = c(4, 4)), row.names = c("Mazda RX4", "Mazda RX4 Wag"
#> ), class = "data.frame")
```

Вы можете поместить полученную структуру прямо в ваш вопрос:

```
example_df <- structure(list(mpg = c(21, 21), cyl = c(6, 6), disp = c(160,
160), hp = c(110, 110), drat = c(3.9, 3.9), wt = c(2.62, 2.875
), qsec = c(16.46, 17.02), vs = c(0, 0), am = c(1, 1), gear = c(4,
4), carb = c(4, 4)), row.names = c("Mazda RX4", "Mazda RX4 Wag"
), class = "data.frame")

example_df
#>      mpg   cyl  disp    hp  drat    wt   qsec    vs  am gear carb
#> Mazda RX4     21     6  160   110   3.9   2.62  16.5  0   1   4   4
#> Mazda RX4 Wag 21     6  160   110   3.9   2.88  17.0  0   1   4   4
```

Вторая часть хорошего воспроизводимого примера – это пример кода. Пример кода должен быть максимально простым и иллюстрировать то, что вы пытаетесь сделать или уже пробовали сделать. Это *не* должен быть большой блок кода со множеством разных вещей. Сделайте так, чтобы пример содержал минимальное количество необходимого кода. Если вы используете какие-либо пакеты, обязательно включите вызов `library` в начале своего кода. Кроме того, не включайте в свой вопрос ничего, что могло бы нанести вред человеку, выполняющему ваш код, например `rm(list = ls())`, что могло бы удалить все объекты R в памяти. Имейте сочувствие к человеку, пытающемуся помочь вам, и поймите, что он добровольно жертвует свое время, чтобы помочь вам, и может выполнять ваш код на той же машине, которую использует для выполнения своей работы.

Чтобы проверить свой пример, откройте новый сеанс R и попробуйте запустить его. После того как вы отредактировали свой код, пришло время дать немного больше информации потенциальным респондентам. В текстовом виде опишите, что вы пытаетесь сделать, что пытались сделать и свой вопрос. Будьте максимально лаконичны. Как и в примере кода, ваша цель – максимально эффективно общаться с человеком, читающим ваш вопрос. Может быть полезно указать в своем описании, какую версию R вы используете и какую платформу (Windows, Mac, Linux). Эту информацию можно легко получить с помощью команды `sessionInfo`.

Если вы собираетесь отправить свой вопрос в список рассылки R, вы должны знать, что на самом деле существует несколько таких списков. R-help – это основной список для общих вопросов. Существует также множество списков рассылки специальных групп интересов (SIG), посвященных конкретным областям, таким как генетика, финансы, разработка на языке R и даже вакансии для разработчиков. Полный список можно увидеть на странице <https://stat.ethz.ch/mailman/listinfo>. Если ваш вопрос относится к какой-то области, вы получите более подходящий ответ, выбрав соответствующий список. Однако, как и в случае с R-help, внимательно выполняйте поиск в архивах списков SIG, прежде чем задавать свой вопрос.

См. также

Мы предлагаем вам прочитать превосходное эссе Эрика Рэймонда и Рика Моена под названием «Как правильно задавать вопросы» (<http://www.catb.org/~esr/faqs/smart-questions.html>), перед тем как отправлять какие-либо вопросы. Серьезно. Прочтите его.

На сайте Stack Overflow есть отличный пост, в котором содержатся подробные сведения о создании воспроизводимого примера. Его можно найти на странице <https://stackoverflow.com/q/5963269/37751>.

У Дженни Брайан есть замечательный пакет под названием `гергех`, который помогает создать хороший воспроизводимый пример и предоставляет вспомогательные функции для написания текста на языке разметки Markdown для таких сайтов, как Stack Overflow. Этот пакет можно найти на ее странице в GitHub (<https://github.com/tidyverse/reprex>).