УДК 004.04
ББК 32.372
К12

Роберт И. Кабаков
К12 R в действии / пер. с англ. А. Н. Киселева. – М.: ДМК Пресс, 2023. – 768 с.: ил.

ISBN 978-5-93700-173-3

R – золотой стандарт, ежедневно используемый исследователями по всему миру для самых разных вычислений и статистического анализа данных. Этот свободно распространяемый язык с открытым исходным кодом включает огромное количество пакетов самой разной направленности, от расширенной визуализации данных до глубокого обучения. Чрезвычайно удобный для пользователей с математическим складом ума, R легко решает практические задачи, не заставляя думать о них с точки зрения программиста.

Данная книга научит вас выполнять статистический анализ и визуализировать результаты с помощью R и его популярных пакетов; решать такие практические задачи, как прогнозирование, интеллектуальный анализ данных и разработка динамических отчетов. В третье издание добавлены новые сведения о построении диаграмм с помощью пакета ggplot2, а также приводятся примеры из области машинного обучения, такие как кластеризация, классификация и анализ временных рядов.

Издание предназначено для широкого круга специалистов по обработке данных.

УДК 004.04
ББК 32.372

ISBN (анг.) 978-1-61729-605-5
ISBN (рус.) 978-5-93700-173-3

Краткое оглавление

Оглавление

Предисловие

Что толку в книжке, если в ней нет ни картинок, ни разговоров?

Алиса. *«Алиса в Стране чудес»*[1]

Оно чудесно и наделено сокровищами, способными удовлетворить всех от мала до велика, но не предназначено для робких духом.

Кью. *Сериал «Звездный путь: следующее поколение»*

Когда я начал писать эту книгу, я потратил довольно много времени на выбор хорошего эпиграфа. В итоге я остановился на этих двух. R – это потрясающе гибкая платформа и язык для исследования, визуализации и интерпретации данных. Я выбрал цитату из «Алисы в Стране чудес», чтобы передать суть современного статистического анализа – интерактивного процесса, состоящего из исследования, визуализации и интерпретации.

Вторая цитата отражает широко распространенное мнение о том, что R сложен в изучении. Я надеюсь показать вам, что это не так. R обладает настолько широкими возможностями и предлагает такое огромное число аналитических и графических функций (по последним подсчетам их более 50 000), что в одинаковой степени может вызывать бессознательный страх и у новичков, и у опытных пользователей. Однако в этом кажущемся безумии есть своя логика и поэзия. Вооружившись руководствами и инструкциями, вы сможете сориентироваться в огромном разнообразии возможностей и выбрать те инструменты, которые нужны для эффективного и элегантного решения вашей задачи.

Первое мое знакомство с R состоялось несколько лет назад, когда я подал заявление о приеме на должность консультанта по статистике. На встрече перед собеседованием будущий работодатель

[1] Перевод Н. Демуровой.

спросил меня, владею ли я языком R. Следуя стандартным советам специалистов по подбору персонала, я немедленно сказал «да» и приступил к его изучению. Я был опытным статистиком и исследователем с 25-летним опытом программирования в SAS и SPSS, свободно владел несколькими языками программирования. Что тут может быть сложного? Знаменитые последние слова.

Стремясь выучить этот язык программирования (как можно быстрее, ведь день собеседования приближался с угрожающей быстротой), я находил или тома, посвященные внутренней структуре языка, или многочисленные трактаты об отдельных продвинутых статистических методах, написанных специалистами в данной области для своих коллег. Встроенная справка была слишком лаконичной и служила скорее справочником, чем учебным пособием. Каждый раз, когда мне казалось, что я освоил общую логику и возможности R, обнаруживалось что-то новое, заставлявшее почувствовать себя невежественным и ничтожным.

Взявшись осваивать R, я подошел к процессу с позиции исследователя данных. Я пытался понять, что нужно сделать, чтобы успешно обработать, проанализировать и интерпретировать данные, и выделил следующие важные аспекты:

- доступ к данным (получение данных из разных источников);
- очистка данных (замена или удаление пропущенных значений, преобразование признаков в более удобный для обработки формат);
- аннотирование данных (чтобы можно было вспомнить, что представляет каждый их фрагмент);
- обобщение данных (вычисление описательных статистик, помогающих характеризовать данные);
- визуализация данных (потому что картинка на самом деле стоит тысячи слов);
- моделирование данных (выявление зависимостей и проверка гипотез);
- оформление результатов (подготовка таблиц и диаграмм достаточного для публикации качества).

Затем я постарался понять, как можно использовать R, чтобы выполнить каждую из этих задач. Поскольку я лучше всего учусь, обучая других, со временем я создал сайт (www.statmethods.net), на котором рассказываю все, что узнал сам.

Затем, спустя год, Марьян Бейс (Marjan Bace) из издательства Manning позвонила и спросила, не хочу ли я написать книгу про R. К этому времени у меня уже было 50 статей в научных журналах, четыре технических руководства, многочисленные главы в книгах и целая книга по методологии исследований, и что тут может быть сложного? Рискую повториться – знаменитые последние слова.

Первое издание вышло в 2011 году, а второе – в 2015-м. Над третьим изданием я начал работать два с половиной года назад. Описание R всегда было непростой задачей, но за последние несколько лет произошла почти что революция, обусловленная ростом популярности больших данных, широким внедрением программного обеспечения tidyverse (tidyverse.org), быстрой разработкой новых подходов к прогнозной аналитике и машинному обучению, а также появлением новых и более мощных технологий визуализации данных. Я хотел отразить все эти важные изменения в третьем издании.

Книгу, которую вы держите в руках, я мечтал иметь много лет назад. Я постарался написать для вас путеводитель по R, который позволит быстро овладеть всеми возможностями этого уникального продукта с открытым исходным кодом, не испытав разочарований и раздражения, которые пришлось испытать мне. Надеюсь, вам понравится.

P.S. Мне предложили ту должность, но я отказался. Однако знакомство с R развернуло мою карьеру в совершенно неожиданном направлении. Жизнь может быть забавной штукой.

Об этой книге

Если вы выбрали эту книгу, скорее всего, у вас есть какие-то данные, которые нужно собрать, обобщить, преобразовать, исследовать, смоделировать, визуализировать или представить коллегам. Если это так, то R создан для вас! R стал всемирно известным языком программирования для статистического анализа и визуализации данных. В нем реализовано множество методов анализа данных, от самых простых до самых сложных и современных.

Как проект с открытым кодом он доступен для многих платформ, включая Windows, Mac OS X и Linux. Он постоянно развивается, и ежедневно появляются новые процедуры. Кроме того, R поддерживается большим и многоликим сообществом ученых и программистов, которые охотно помогут новичку советами.

Платформа R больше, пожалуй, известна за способность создавать красивые и сложные диаграммы, она может справиться с любой статистической задачей. Базовая версия содержит сотни функций для статистического анализа, управления данными и построения диаграмм. Однако некоторые особенно мощные методы реализованы в дополнительных пакетах, созданных независимыми авторами.

Эта широта возможностей имеет свою цену. Новичкам порой сложно понять, что такое R и как работать с этим языком. Даже самые опытные пользователи R с удивлением обнаруживают какие-то возможности, о которых не подозревали.

Третье издание «R в действии» – это руководство-путеводитель по R, знакомящее с самой платформой и ее возможностями. В книге описаны наиболее полезные функции базовой версии и более 90 наиболее часто используемых дополнительных пакетов. Основной упор в книге делается на практическое применение – на то, чтобы вы, руководствуясь прочитанным, могли проанализировать

ваши данные и изложить результаты коллегам. По окончании чтения этой книги вы будете иметь хорошее представление о том, как работает R и где можно получить дополнительную информацию. Вы научитесь применять разнообразные методы визуализации данных и обретете достаточно умений, чтобы справиться как с простыми, так и со сложными задачами анализа данных.

Что нового в третьем издании

В третье издание внесены многочисленные изменения, в том числе широко освещаются приемы применения tidyverse для управления данными и их анализа. Вот некоторые из наиболее заметных изменений.

Глава 2 (создание набора данных) теперь включает описание пакетов `readr`, `readxl` и `haven`, реализующих импорт данных. Также появился новый раздел о `tibbles`, современном решении поддержки наборов данных.

Главы 3 (основы управления данными) и 5 (дополнительные приемы управления данными) включают описание пакетов `dplyr` и `tidyr`, предназначенных для управления данными, их преобразования и обобщения.

Главы 4 (начало работы с диаграммами), 6 (базовые диаграммы), 11 (диаграммы средней сложности) и 19 (продвинутые методы работы с диаграммами) переписаны заново и подробно рассказывают о пакете `ggplot2` и его возможностях.

Глава 16 (кластерный анализ) описывает улучшенные приемы создания диаграмм и содержит новый раздел, посвященный оценке возможности кластеризации данных.

Глава 17 (классификация) содержит новый раздел, посвященный использованию иерархических диаграмм и диаграмм значений Шепли, помогающих в создании моделей черного ящика.

Глава 18 (продвинутые методы работы с пропущенными данными) была дополнена новыми разделами о методах *k*-ближайших соседей и случайного леса для подстановки отсутствующих значений.

Глава 20 (продвинутые приемы программирования) содержит новые разделы, посвященные нестандартным способам вычислений и визуальной отладке.

Глава 21 (создание динамических отчетов) приводит расширенное описание R Markdown и содержит новые разделы, посвященные параметризованным отчетам и распространенным ошибкам программирования.

Глава 22 (создание пакета) была полностью переписана и теперь включает описание способов использования новых инструментов для упрощенного создания пакетов, а также включает новые разделы о приемах распространения своих пакетов через CRAN, GitHub и веб-сайты.

Приложение A (графические пользовательские интерфейсы) было обновлено, чтобы отразить быстрые изменения в этой области.

Приложение B (настройка среды выполнения) было пересмотрено и теперь описывает новые методы настройки и рассказывает о влиянии потенциальных побочных эффектов на воспроизводимость исследований.

Приложение F (работа с большими наборами данных) содержит информацию о новых пакетах для работы с большими наборами данных, объем которых превышает объем ОЗУ, аналитических методах анализа наборов данных терабайтного размера и применении R в облачных службах.

На протяжении всей книги вам будут встречаться новые разделы, описывающие приемы использования RStudio для программирования, отладки и создания отчетов и пакетов. Наконец, в текст были внесены многочисленные обновления и исправления.

Кому адресована эта книга

Третье издание книги «R в действии» предназначено для всех, кто имеет дело с данными. Опыт в программировании статистических методов не требуется. Хотя эта книга доступна и новичкам, в ней содержится достаточно нового и полезного материала даже для опытных специалистов по R.

Пользователи, не владеющие познаниями в области статистики, но желающие использовать R для управления данными, их обобщения и представления в графическом виде, без особого труда освоят главы 1–6, 11 и 19. Главы 7 и 10 предполагают наличие у читателя базовых знаний математической статистики, а главы 8, 9 и 12–18 потребуют более глубоких познаний в этой области. Однако я старался писать каждую главу так, чтобы в ней было что-то интересное и полезное и для новичков, и для опытных статистиков.

Структура книги

Эта книга создана как путеводитель по R, с упором на методы, которые можно сразу применить для управления данными, их визуализации и анализа. Книга состоит из 22 глав, сгруппированных в четыре части: «Начало работы», «Базовые методы», «Методы средней сложности» и «Методы повышенной сложности». Дополнительные темы рассмотрены в семи приложениях.

Глава 1 начинается с общего обзора особенностей R, делающих его столь полезным для обработки данных. В главе рассказано, как установить платформу R и расширить ее возможности установкой дополнительных пакетов. Оставшаяся часть главы посвящена описанию интерфейса и способов запуска ее в интерактивном и пакетном режимах.

В главе 2 описаны многие методы импорта данных. Первая половина главы посвящена представлению структур, предназначенных для хранения данных в R. Во второй половине главы рассказывается о способах ввода данных с клавиатуры, импорта из текстовых файлов, веб-страниц, электронных таблиц, из других статистических программ и баз данных.

Глава 3 посвящена основам управления данными, включая сортировку, объединение и разбиение, а также преобразование, перекодировку и удаление переменных.

Глава 4 знакомит с синтаксисом создания диаграмм для визуализации данных. Здесь мы обсудим методы создания диаграмм, их изменения и сохранения в разных форматах.

Глава 5 основана на главах 3 и 4 и содержит описание функций (математических, статистических, текстовых) и управляющих конструкций (циклы, условное выполнение) для управления данными. Затем мы поговорим о том, как написать свою функцию на R и как сгруппировать данные различными способами.

Глава 6 рассказывает о создании наиболее распространенных одномерных диаграмм, таких как столбиковая и круговая диаграммы, диаграмма распределения плотности, диаграмма размахов (коробчатая диаграмма) и точечная диаграмма. Все эти диаграммы помогают изучать характер распределения значений одной переменной.

Глава 7 посвящена обобщению данных, включая использование описательных статистик и сводных таблиц. Затем в ней рассматриваются основные способы анализа взаимосвязей между двумя переменными, включая корреляцию, *t*-критерий Стьюдента, критерий хи-квадрат и непараметрические методы.

Глава 8 знакомит с применением методов регрессионного анализа для моделирования взаимосвязей между числовой переменной-откликом (outcome variable) и набором из одной или нескольких независимых переменных (predictor variables). Подробно рассмотрены методы подгонки этих моделей, оценки их адекватности и интерпретации.

В главе 9 представлены основы дисперсионного анализа и его разновидности. Обычно дисперсионный анализ выполняется с целью выяснить, как комбинации разных типов воздействий или разных условий влияют на числовую переменную-отклик. Также описаны методы оценки адекватности анализа и визуализации результатов.

Глава 10 подробно описывает анализ мощности статистических критериев. Она начинается с обсуждения проверки гипотез; затем мы поговорим о том, как определить объем выборки, необходимый для выявления влияния размера на заданный уровень достоверности. Это поможет вам повысить вероятность достижения желаемого результата при планировании экспериментов.

Глава 11 продолжает обсуждение, начатое в главе 5. В ней рассказано, как создать диаграммы для визуализации связей между двумя и более переменными. Рассматриваются разные типы двух- и трехмерных диаграмм рассеяния, матриц диаграмм рассеяния, графиков, коррелограмм и мозаичных диаграмм.

В главе 12 представлены аналитические методы для обработки данных, полученных из источников с неизвестными или смешанными распределениями, когда размеры выборок малы, когда часто встречаются выбросы или когда разработка статистического критерия на основании наблюдаемого распределения слишком сложна, в том числе метод повторной выборки (resampling) и бутстреп-анализ (bootstrapping), требующие большого объема вычислений и легко реализуемые в R.

Глава 13 вновь возвращается к обсуждению регрессионного анализа, начатому в главе 8, и охватывает подходы к анализу данных с распределением, отличным от нормального. Глава начинается с описания обобщенных линейных моделей. Затем более подробно рассматриваются случаи, когда нужно предсказать переменную-отклик, представленную либо категориальными (логистическая регрессия), либо счетными данными (пуассоновская регрессия).

Одна из сложностей, связанных с многомерными данными, – задача снижения их размерности. В главе 14 описаны методы, с помощью которых большое число коррелирующих друг с другом переменных преобразуются в меньший набор независимых переменных (метод главных компонент), а также методы выявления скрытой структуры в имеющемся наборе переменных (факторный анализ). Детально разобраны многочисленные этапы этих типов анализа.

Глава 15 описывает методы создания, обработки и моделирования временных рядов, визуализацию и разложение временных рядов, а также экспоненциальный подход и подход на основе интегрированной модели авторегрессии скользящего среднего (AutoRegressive Integrated Moving Average, ARIMA) к прогнозированию будущих значений.

Глава 16 иллюстрирует методы кластеризации наблюдений в естественные группы. Она начинается с обсуждения общих этапов комплексного кластерного анализа, а затем переходит к представлению методов иерархической кластеризации и сегментирования. Здесь мы рассмотрим несколько методов определения оптимального количества кластеров.

В главе 17 представлены популярные методы машинного обучения с учителем для классификации наблюдений по группам. По очереди будут рассмотрены деревья решений, случайные леса и метод опорных векторов. Здесь вы также узнаете о методах оценки точности каждого подхода и новых приемах анализа результатов.

Следуя стремлению познакомиться с наиболее актуальными методами анализа данных, в главе 18 мы поговорим о современных подходах к решению распространенной проблемы пропущенных значений в данных. В R реализованы разнообразные изящные подходы к анализу неполных в силу разных причин данных. Здесь описаны лучшие из этих методов и разъясняется, когда и какие стоит применять, а каких лучше избегать.

Глава 19 завершает обсуждение диаграмм знакомством с некоторыми наиболее сложными и полезными методами настройки осей координат, применения цветовых схем, шрифтов, легенд и аннотаций. Вы узнаете, как объединить несколько диаграмм в одну и, наконец, как превратить статический график в интерактивную веб-визуализацию.

Глава 20 рассматривает передовые методы программирования. Здесь вы познакомитесь с методами объектно-ориентированного программирования и приемами отладки. В этой главе также даются советы по эффективному программированию. Она будет особенно полезна тем, кто желает лучше понять, как работает R, и ее обязательно нужно прочитать, прежде чем переходить к главе 22.

В главе 21 описывается несколько методов создания красивых отчетов на R. Вы узнаете, как создавать веб-страницы, отчеты, статьи и даже книги из программ на R. Полученные документы могут включать код, таблицы результатов, графики и комментарии.

Наконец, в главе 22 представлено пошаговое руководство по созданию пакетов R. Это позволит вам писать сложные программы, эффективно документировать их и делиться ими с другими. Здесь подробно обсуждаются методы распространения и продвижения ваших пакетов.

В послесловии перечислены многие из лучших сайтов, которые следует посетить, чтобы научиться работать с R, влиться в сообщество пользователей R, получить ответы на возникшие вопросы и отслеживать изменения в этом стремительно развивающемся программном продукте.

И последнее, но не менее важное: семь приложений (от A до G) содержат дополнительные сведения по таким полезным темам, как графический пользовательский интерфейс R, настройка и обновление платформы, экспорт данных в другие приложения, использование R для матричных вычислений алгебры (по образцу MATLAB) и работа с большими наборами данных.

Мы также предлагаем дополнительную главу, доступную в интернете на сайте издательства по адресу https://www.manning.com/books/r-in-action-third-edition. Онлайн-глава 23 посвящена пакету `lattice`, предлагающему альтернативный подход к визуализации данных в R.

Совет специалистам по интеллектуальному анализу данных

Сфера интеллектуального анализа данных неразрывно связана с выявлением закономерностей в больших наборах данных. Многие специалисты по интеллектуальному анализу используют R, так как он обладает мощными аналитическими возможностями. Если вы занимаетесь анализом данных и желаете как можно быстрее освоить R, я рекомендую следующую последовательность чтения: глава 1 (введение), глава 2 (структуры данных и разделы, описывающие приемы импорта данных, имеющие к вам прямое отношение), глава 4 (основы управления данными), глава 7 (описательная статистика), глава 8 (разделы 1, 2 и 6, регрессия), глава 13 (раздел 2, логистическая регрессия), глава 16 (кластеризация), глава 17 (классификация) и приложение F (работа с большими наборами данных). Затем читайте другие главы по мере необходимости.

Примеры

Чтобы сделать книгу максимально полезной, я выбрал примеры из разных областей знаний, включая психологию, социологию, медицину, биологию, бизнес и технические науки. Ни один из примеров не требует специальных знаний в соответствующей области.

Наборы данных, используемые в этих примерах, были выбраны потому, что они позволяют формулировать интересные вопросы и имеют небольшой размер. Это позволяет сосредоточиться на рассматриваемом методе и быстро понять происходящее. Когда учишься новым методам, меньше – значит лучше. Наборы данных либо входят в состав дистрибутива R, либо доступны в виде дополнительных пакетов, которые можно скачать из интернета.

Принятые обозначения

В книге использованы следующие типографские обозначения:

- `моноширинный шрифт` – для программного кода, который нужно вводить именно так, как указано в книге;
- `моноширинный шрифт` также используется в основном тексте для обозначения фрагментов кода или ранее упомянутых объектов;
- *`курсив`* внутри программного кода указывает места заполнения. Его следует заменять подходящим текстом или значениями, соответствующими задаче. Например, *`путь_к_моему_файлу`* должен быть заменен путем к реальному файлу на вашем компьютере;
- R – это интерактивный язык, который информирует пользователя о готовности принять команду приглашением (> по умолчанию). Многие фрагменты программного кода в кни-

ге скопированы из интерактивных сеансов. Если вы видите строки кода, начинающиеся с >, не набирайте этот символ приглашения к вводу команды;

- пояснения к программному коду приведены в виде комментариев в тексте. В дополнение к этому некоторые пояснения обозначены нумерованными кружками, такими как ❶, которые отсылают к объяснению ниже в тексте;
- для экономии места или чтобы сделать текст более понятным, кое-где в вывод результатов интерактивных сеансов добавлены дополнительные пробелы или удален текст, который напрямую не относится к обсуждаемой теме.

Фрагменты выполняемого кода доступны в liveBook – онлайн-версии этой книги по адресу https://livebook.manning.com/book/r-in-action-Third-edition. Полный код всех примеров доступен для загрузки на сайте издательства Manning по адресу https://www.manning.com/books/r-in-action-third-edition и на GitHub, по адресу www.github.com/rkabacoff/RiA3. Чтобы получить максимум выгоды от этой книги, я рекомендую опробовать примеры по мере встречи с ними.

Наконец, существует распространенное правило, которое гласит, что если спросить двух статистиков, как проанализировать набор данных, то они дадут три ответа. Оборотная сторона этого утверждения: каждый ответ будет приближать вас к пониманию данных. Я не утверждаю, что тот или иной вид анализа является лучшим или единственным подходом к конкретной задаче. Используя навыки, полученные в этой книге, вы сможете поэкспериментировать с данными и посмотреть, что можно выяснить. R – интерактивный инструмент, предлагающий лучший способ обучения – экспериментирование.

Живое обсуждение книги

Приобретая книгу «R в действии», вы получаете бесплатный доступ к форуму, поддерживаемому издательством Manning Publications, где вы можете оставлять свои комментарии к книге, задавать технические вопросы и отвечать на них, а также получать помощь от автора и других пользователей. Чтобы получить доступ к форуму и зарегистрироваться на нем, откройте в веб-браузере страницу https://livebook.manning.com/book/r-in-action-third-edition/discussion. Узнать больше о форумах Manning и познакомиться с правилами поведения можно по адресу https://livebook.manning.com/#!/discussion.

Издательство Manning обязуется предоставить своим читателям место встречи, где может состояться содержательный диалог между отдельными читателями и между читателями и автором.

Но со стороны авторов отсутствуют какие-либо обязательства уделять форуму какое-то определенное внимание – их присутствие на форуме остается добровольным (и неоплачиваемым). Мы предлагаем задавать авторам стимулирующие вопросы, чтобы их интерес не угасал! Форум и архив с предыдущими обсуждениями остаются доступными на сайте издательства, пока книга продолжает издаваться.

Об авторе

Доктор наук Роберт Кабаков (Robert Kabacoff) – профессор вычислительной аналитики в Уэслианском университете (Wesleyan University) и опытный специалист по анализу данных с более чем 30-летним опытом программирования статистических вычислений и анализа данных в бизнесе, здравоохранении и государственных учреждениях. Вел курсы по анализу данных и статистическому программированию как для студентов, так и для аспирантов, а также поддерживает сайт Quick-R (statmethods.net) и сайт по визуализации данных с помощью R (rkabacoff.github.io/datavis).

Часть I

Начало работы

Добро пожаловать в «R в действии»! R – одна из наиболее популярных платформ для анализа данных и их визуализации из имеющихся в настоящее время. Это бесплатное программное обеспечение, распространяемое с открытым исходным кодом и способное работать в Windows, Mac OS X и Linux. Благодаря этой книге вы приобретете навыки, необходимые для овладения данной многофункциональной платформой и научитесь эффективно применять ее для обработки данных.

Книга разделена на четыре части. Первая часть посвящена установке базовой версии, знакомству с интерфейсом, импорту данных и преобразованию их в удобный для дальнейшего анализа вид. Глава 1 познакомит вас с окружением R. Сначала будет дан краткий обзор платформы R и ее особенностей, которые делают ее столь мощным инструментом для современного анализа данных. После краткого объяснения, как получить и установить R, последует описание пользовательского интерфейса на ряде простых примеров. Затем вы узнаете, как расширить функциональность базовой версии с помощью *дополнительных пакетов*, которые можно найти в репозиториях в интернете. И в конце главы приводится пример, на котором вы сможете проверить ваши новые умения.

После знакомства с интерфейсом R встает следующая задача – загрузить данные в программу. В современном богатом информацией мире данные могут поступать из разных источников и в разных форматах. В главе 2 описано множество методов импорта данных в R. В первой половине главы мы поговорим о форматах, в которых R может хранить данные, и как можно вводить данные вручную. Во второй части обсуждаются методы импорта данных из текстовых файлов, веб-страниц, электронных таблиц, других статистических программ и баз данных.

Данные редко поступают в формате, пригодном для использования. Зачастую приходится потратить немало времени, чтобы объединить данные из разных источников, очистить от ошибочных данных (неверно закодированные или несоответствующие данные, а также пропуски) и создать новые переменные (комбинированные, преобразованные или перекодированные), прежде чем можно будет приступать к поиску ответов на поставленные вопросы. В главе 3 описаны все основные способы управления данными в R, включая сортировку, объединение и сегментирование наборов данных, а также преобразование, перекодировку и удаление переменных.

Многие пользователи, впервые познакомившиеся с R, больше интересуются ее мощными графическими возможностями. Поэтому в главе 4 будет дан обзор *грамматики диаграмм* – пакета `ggplot2`. Для начала мы построим простую диаграмму, а затем будем последовательно расширять ее возможности, пока не получим комплексную визуализацию данных. В этой главе вы также узнаете, как задать основные настройки диаграммы и сохранить ее в различных графических форматах.

Глава 5 основана на сведениях, изложенных в главе 3. Она рассказывает, как использовать числовые (арифметические, тригонометрические, статистические) и текстовые функции (разбиение строк, конкатенация, замена) для управления данными. Для иллюстрации описываемых функций в этом разделе приводится множество примеров. Затем мы обсудим управляющие конструкции (циклы, условные операторы). Прочитав этот раздел, вы научитесь определять свои функции на R. Это позволит расширить возможности R, объединив многие команды в одну легко настраиваемую функцию. В заключение обсуждаются мощные методы реорганизации и группировки данных, которые часто бывают полезными при подготовке данных к дальнейшему анализу.

К концу части I вы будете готовы начать программировать в среде R. Приобретете навыки, необходимые для ввода данных и получения их из внешних источников, а также для их очистки. Кроме того, вы получите опыт создания, настройки и сохранения различных типов диаграмм.

1 Знакомство с R

В этой главе:

- установка R и RStudio;
- знакомство с языком программирования R;
- запуск программ.

В последние годы подходы к анализу данных принципиально изменились. С появлением персональных компьютеров и интернета объемы данных значительно возросли. Коммерческие компании обладают терабайтами данных о потребителях, правительственные, академические и частные исследовательские институты оперируют обширными архивными данными и материалами обследований по многим направлениям. Извлечение информации (не говоря уже о знаниях) из этих огромных объемов данных превратилось в самостоятельную отрасль. В то же время задача представления информации в легкодоступном и усвояемом виде значительно усложнилась.

Развитие наук, посвященных анализу данных (статистика, психометрика, эконометрика, машинное обучение), не отстает от взрывообразного роста объема данных. До эпохи персональных компьютеров и интернета новые статистические методы разрабатывались учеными-теоретиками, которые публиковали свои результаты в виде статей в специализированных журналах. Могли

пройти годы, прежде чем эти методы доходили до программистов и встраивались в программы статистической обработки данных. В наше время новые методы появляются *ежедневно*. Исследователи-статистики публикуют новые и усовершенствованные методы вместе с программным кодом, который их реализует, на общедоступных веб-сайтах.

Появление персональных компьютеров повлияло на подходы к анализу данных еще с одной стороны. Когда для анализа данных использовались большие ЭВМ, время их работы стоило дорого и его не хватало на всех. Поэтому аналитикам приходилось заранее тщательно определять все параметры анализа. Когда анализ завершался, результаты распечатывались на десятках или сотнях страниц. Аналитик должен был просмотреть их все, оставляя нужное и отсеивая лишнее. В ту пору появились многие популярные статистические пакеты (такие как SAS и SPSS), которые продолжают следовать этому алгоритму до сих пор.

С появлением недорогих и доступных персональных компьютеров произошла смена парадигмы. Теперь процесс анализа не требует предварительной установки всех параметров анализа и стал в значительной степени интерактивным. При этом результаты одного этапа анализа используются как исходные данные для следующего. На рис. 1.1 показана типичная схема анализа данных. На любом этапе анализа могут выполняться преобразование данных, вставка пропущенных значений, добавление или удаление переменных, после чего процесс продолжается. Завершается этот процесс, когда аналитик посчитает, что он или она полностью исследовал(а) данные и ответил(а) на все относящиеся к делу вопросы, на которые можно было ответить.

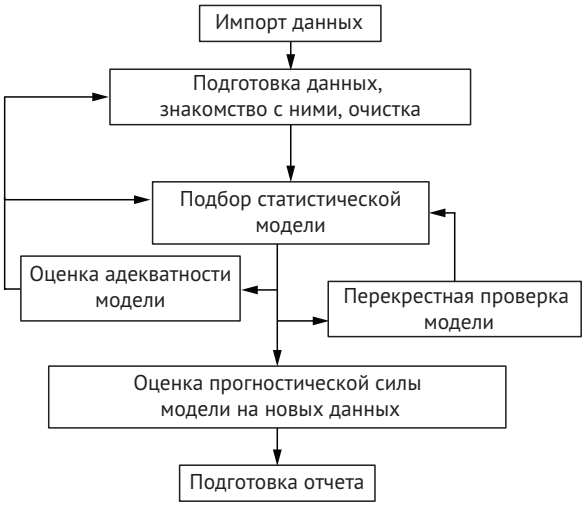

Рис. 1.1. Этапы типичного анализа данных

Появление персональных компьютеров (и особенно мониторов с высоким разрешением) также повлияло на способы представления результатов и их изучение. Изображение действительно может стоить тысячи слов, а люди весьма успешно извлекают информацию из визуальных образов. Современные методы обработки данных все больше опираются на графические способы представления результатов.

Современному исследователю необходимо получать данные из разных источников (систем управления базами данных, текстовых файлов, статистических программ и электронных таблиц), объединять фрагменты данных, маркировать их и очищать от ошибок, анализировать с помощью новейших методов, представлять результаты в наглядном и простом для интерпретации виде, а также включать результаты в привлекательные отчеты, которые не стыдно будет распространять среди заинтересованных лиц и общественности.

1.1. Зачем использовать R?

R – это язык программирования и среда для статистических вычислений и графического анализа, сходный с языком S, первоначально разработанным в Bell Labs. Это инструмент с открытым исходным кодом для анализа данных, который поддерживается большим и активным сообществом исследователей по всему миру. Однако существует много распространенных программ для статистической и графической обработки данных (таких как Microsoft Excel, SAS, IBM SPSS, Stata и Minitab). Так в чем преимущества R?

R обладает множеством уникальных особенностей, которые позволяют рекомендовать именно эту платформу:

- большинство коммерческих статистических программ стоят тысячи, если не десятки тысяч долларов. R – бесплатный программный продукт! Если вы преподаватель или студент, то выгода очевидна;
- R – мощная среда для статистических вычислений, в которой реализованы все основные виды анализа данных;
- в R реализованы сложные статистические процедуры, недоступные в других программах. На самом деле новые функции появляются еженедельно. Если используете SAS, то знаете, насколько маловероятным выглядит появление новых SAS PROC каждые несколько дней;
- R имеет современные графические возможности. Если вам нужно визуализировать сложные данные, то учтите, что в R

реализованы самые разнообразные и мощные методы анализа данных из доступных;

- R – мощная платформа для интерактивного анализа и исследования данных. С самого начала она разрабатывалась для поддержки подхода, показанного на рис. 1.1. Например, результаты любого этапа анализа можно сохранить, обработать и использовать в роли исходных данных для дополнительного анализа;
- зачастую невозможно получить данные из разных источников в пригодном для анализа виде. R может импортировать данные из самых разных источников, включая текстовые файлы, системы управления базами данных, другие статистические программы и специализированные хранилища данных, а также экспортировать данные в форматах всех этих систем;
- R предоставляет непревзойденную платформу, упрощающую программирование новых статистических методов. Она легко расширяется и имеет удобный и естественный язык, позволяющий, например, быстро запрограммировать недавно опубликованный метод;
- возможности R можно интегрировать в приложения, написанные на других языках, включая C++, Java, Python, PHP, Pentaho, SAS и SPSS. Это позволяет продолжать работать на знакомом языке, добавляя возможности R в приложения;
- R поддерживает разные операционные системы, включая Windows, Unix и Mac OS X. Эту платформу можно запустить практически на любом компьютере (я даже видел руководства по установке R на iPhone, что впечатляет, но вряд ли является хорошей идеей);
- для тех, кто не желает учить новый язык, существует множество графических пользовательских интерфейсов, в которых мощь R реализована в форме меню и диалогов.

Демонстрацию графических возможностей R можно видеть на рис. 1.2. На этой диаграмме показана взаимосвязь между стажем работы и заработной платой мужчин и женщин в шести отраслях, на основе данных, полученных в ходе обследования населения США в 1985 году. Технически это матрица диаграмм рассеяния, где пол отображается цветом и символом. Тенденции описываются с помощью линейной регрессии. Если эти термины, *диаграмма рассеяния* и *линейная регрессия*, вам незнакомы, не волнуйтесь. Мы рассмотрим их в последующих главах.

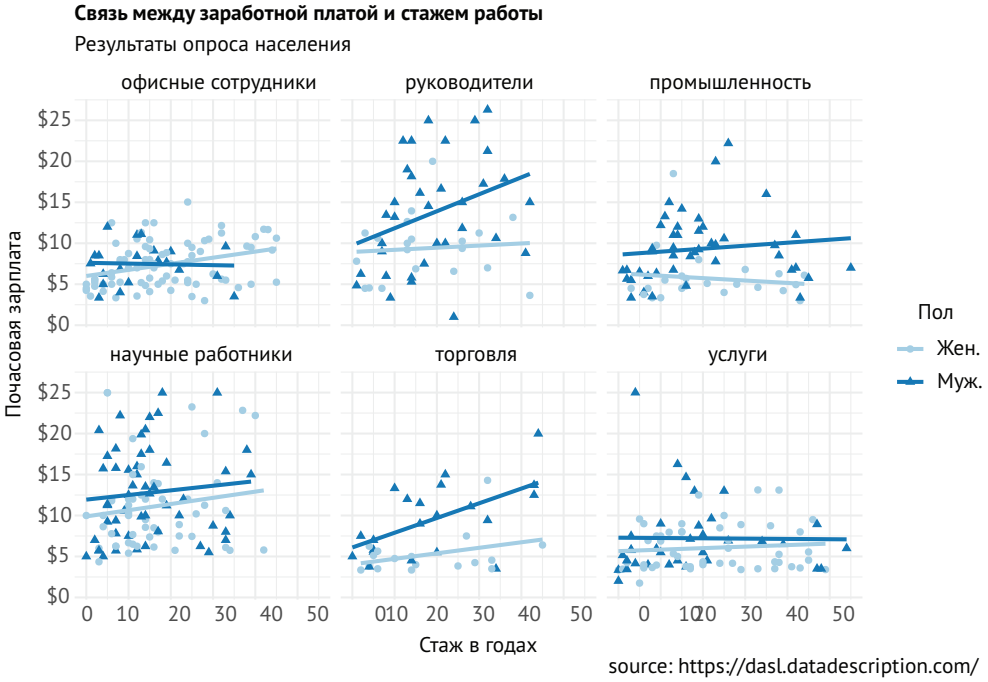

Рис. 1.2. Взаимосвязь между стажем работы и заработной платой мужчин и женщин в шести отраслях. Диаграммы, подобные этой, в R создаются всего несколькими строками кода (эта диаграмма создана с помощью пакета mosaicData)

Вот главное, что показывает этот график:

- взаимосвязи между стажем работы и заработной платой различаются в зависимости от пола и отрасли;
- в сфере услуг заработная плата мало зависит от стажа работы для обоих полов;
- на руководящих должностях заработная плата растет пропорционально стажу работы для мужчин, но не для женщин.

Являются ли эти различия реальными или их можно объяснить случайной изменчивостью выборки? Мы обсудим это далее в главе 8. Но важно отметить, что R позволяет создавать красивые и информативные графики простым и понятным способом. Создание подобных графиков в других статистических программах – сложная, а подчас невыполнимая задача.

К сожалению, для языка R свойственна крутая кривая обучения. Вследствие богатейшего набора возможностей объем документации и файлов справки очень велик. Кроме того, многие из этих возможностей реализованы в дополнительных модулях, созданных независимыми исследователями, поэтому справочная до-

кументация бывает разобщенной и труднодоступной. Научиться выполнять все виды анализа, реализованные в R, действительно очень непросто.

Задача этой книги – сделать овладение R быстрым и простым. Мы рассмотрим многие возможности R, чтобы вы смогли начать обработку своих данных, и подскажем, где можно получить дополнительную информацию. Начнем с установки программы.

1.2. Получение и установка R

R можно бесплатно скачать из «всеобъемлющего сетевого архива R» (Comprehensive R Archive Network, CRAN) по адресу http://cran.r-project.org. Предварительно скомпилированные файлы доступны для Linux, Mac OS X и Windows. Просто следуйте инструкциям по установке для вашей операционной системы. Позже мы обсудим, как расширить возможности R при помощи дополнительных модулей, называемых пакетами, – они также доступны в CRAN.

1.3. Работа в R

R – это чувствительный к регистру символов интерпретирующий язык программирования. Команды можно вводить по одной в строке приглашения к вводу (>) или запускать наборы команд, перечисленные в файлах. Типы данных очень разнообразны: векторы, матрицы, таблицы данных (похожи на наборы данных) и списки (коллекции объектов). Мы обсудим все эти типы данных в главе 2.

Основные функциональные возможности R реализуются при помощи встроенных и пользовательских функций, которые создают и управляют объектами. *Объектами* называются программные структуры, способные хранить значения. В R объектами является вообще все (данные, функции, диаграммы, результаты анализа и т. д.). У каждого объекта есть *атрибут класса* (один или несколько текстовых описателей), который определяет, как выводить, отображать, обобщать или как-то иначе управлять объектом.

В течение интерактивного сеанса все объекты хранятся в памяти. Основные функции доступны по умолчанию. Другие функции содержатся в пакетах, которые при необходимости следует подключать к текущему сеансу.

Инструкции состоят из имен функций и операторов присваивания. Для обозначения присваивания в R используется символ <- вместо привычного =.

Например, инструкция

```
x <- rnorm(5)
```

создает объект вектора с именем x, содержащий пять случайных значений из нормального распределения.

ПРИМЕЧАНИЕ. R позволяет использовать знак равенства = для присваивания. Однако очень немногие функции написаны с его использованием. Это нестандартный синтаксис, поэтому в некоторых ситуациях он не будет работать, и программисты на R поднимут вас на смех. Операцию присваивания допускается записывать в обратном порядке. Например, выражение `rnorm (5) -> x` эквивалентно предыдущей инструкции. И снова, так писать не принято, и я не рекомендую использовать такой стиль.

Комментарии начинаются с символа #. Любой текст после # игнорируется программой. Пример программы с комментариями можно увидеть в разделе 1.3.1.

1.3.1. Начало работы

Первый шаг на пути к использованию R – это, конечно же, установка. Инструкции по установке вы найдете на сайте CRAN. После установки запустите R. Если вы используете Windows, запустите R из меню Start (Пуск). В операционной системе Mac дважды щелкните на значке R в папке Applications (Приложения). В Linux введите команду R в командной строке. Любое из этих действий запустит R (см. рис. 1.3 в качестве примера).

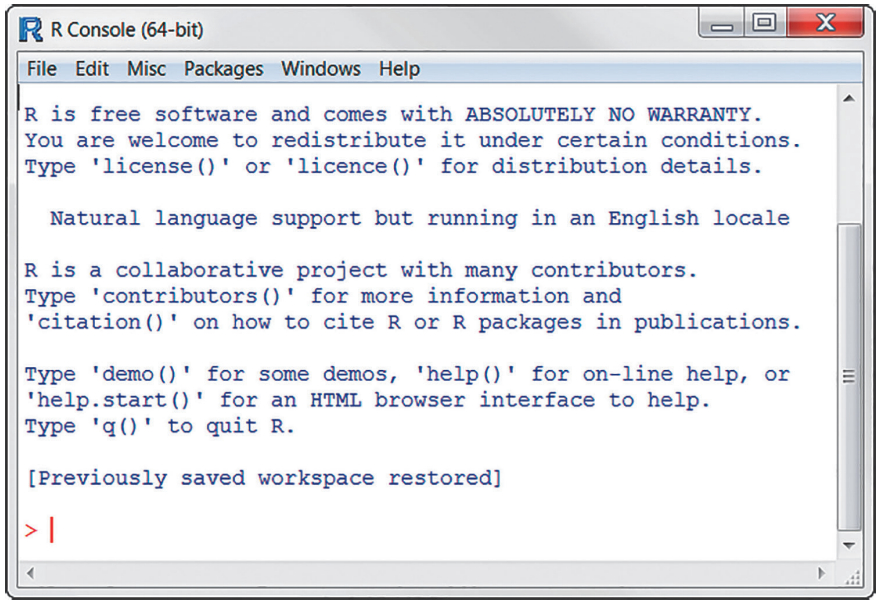

Рис. 1.3. Пример интерфейса R в операционной системе Windows

Чтобы освоиться с интерфейсом, давайте потренируемся на простом выдуманном примере. Представьте, что вы изучаете физическое развитие и собрали данные о возрасте и весе 10 младен-

цев первого года жизни (табл. 1.1). Вам интересно узнать закономерность распределения веса в зависимости от возраста.

Таблица 1.1. Возраст и вес 10 младенцев

Возраст (месяцы)	Вес (кг)	Возраст (месяцы)	Вес (кг)
01	4,4	09	7,3
03	5,3	03	6,0
05	7,2	09	10,4
02	5,2	12	10,2
11	8,5	03	6,1

В листинге 1.1 показан сеанс анализа. Вес и возраст вводятся в виде векторов с помощью функции `c()`, которая преобразует свои аргументы в вектор или список. Затем вычисляется среднее арифметическое и стандартное отклонение для значений веса, а также коэффициент корреляции между возрастом и весом, которые вычисляются с помощью функций `mean()`, `sd()` и `cor()` соответственно. После этого с помощью функции `plot()` отображается диаграмма, отражающая зависимость веса от возраста и позволяющая визуально наблюдать тенденцию. Функция `q()` завершает сеанс.

Листинг 1.1. Пример сеанса работы с R

```
> age <- c(1,3,5,2,11,9,3,9,12,3)
> weight <- c(4.4,5.3,7.2,5.2,8.5,7.3,6.0,10.4,10.2,6.1)
> mean(weight)
[1] 7.06
> sd(weight)
[1] 2.077498
> cor(age,weight)
[1] 0.9075655
> plot(age,weight)
```

Как можно видеть в листинге 1.1, средний вес этих 10 младенцев составляет 7,06 кг, а стандартное отклонение равно 2,08 кг и существует сильная линейная взаимосвязь между возрастом и весом (коэффициент корреляции равен 0,91). Эта взаимосвязь также видна на диаграмме рассеяния на рис. 1.4. Неудивительно, что по мере взросления младенцы в среднем становятся тяжелее.

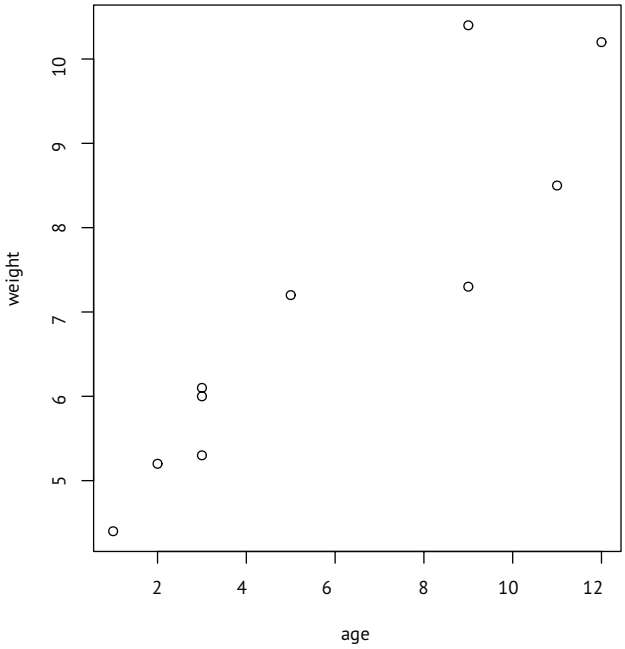

Рис. 1.4. Диаграмма рассеяния веса младенцев (weight, в килограммах) в зависимости от их возраста (age, в месяцах)

Диаграмма рассеяния на рис. 1.4 достаточно информативна, но выглядит не очень привлекательно. В последующих главах вы узнаете, как создавать более привлекательные и сложные диаграммы.

> **СОВЕТ.** Чтобы получить представление о графических возможностях R, взгляните на примеры, представленные в разделе «Data Visualization with R» в документации (http://rkabacoff.github.io/datavis) и в статье «The Top 50 ggplot2 Visualizations – The Master List» (http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html).

1.3.2. Использование RStudio

Стандартный интерфейс R очень прост и предлагает ничем не примечательную командную строку для ввода инструкций. Для создания реальных проектов лучше использовать более полноценный инструмент, помогающий писать код и просматривать результаты. Для R было создано несколько таких инструментов, которые называют интегрированными средами разработки (Integrated Development Environment, IDE), включая Eclipse с расширением StatET, Visual Studio for R и RStudio Desktop.

RStudio Desktop (https://www.rstudio.com) – самый, пожалуй, популярный выбор. Эта среда разработки предлагает многооконный интерфейс с вкладками и инструментами для импорта данных, написания кода, отладки ошибок, визуализации вывода и создания отчетов.

RStudio распространяется бесплатно, как продукт с открытым исходным кодом, и поддерживает Windows, Mac и Linux. Поскольку RStudio является интерфейсом к R, перед установкой RStudio Desktop следует обязательно установить R.

> **СОВЕТ.** Для настройки интерфейса RStudio выберите меню Tools > Global Options... (Инструменты > Общие настройки...). На вкладке General (Общие) я рекомендую снять флажок Restore .RData into Workspace at Startup (Восстанавливать .RData в рабочей области при запуске) и выбрать значение Never (Никогда) для параметра Save Workspace to .RData on Exit (Сохранять рабочую область в .RData при выходе). Это обеспечит получение чистого окружения при каждом запуске RStudio.

Давайте повторно запустим код из листинга 1.1, но теперь в RStudio. В Windows запустите RStudio из меню Start (Пуск). В Mac дважды щелкните на значке RStudio в папке Applications (Приложения). В Linux введите команду `rstudio` в командной строке. На всех трех платформах появится один и тот же интерфейс (рис. 1.5).

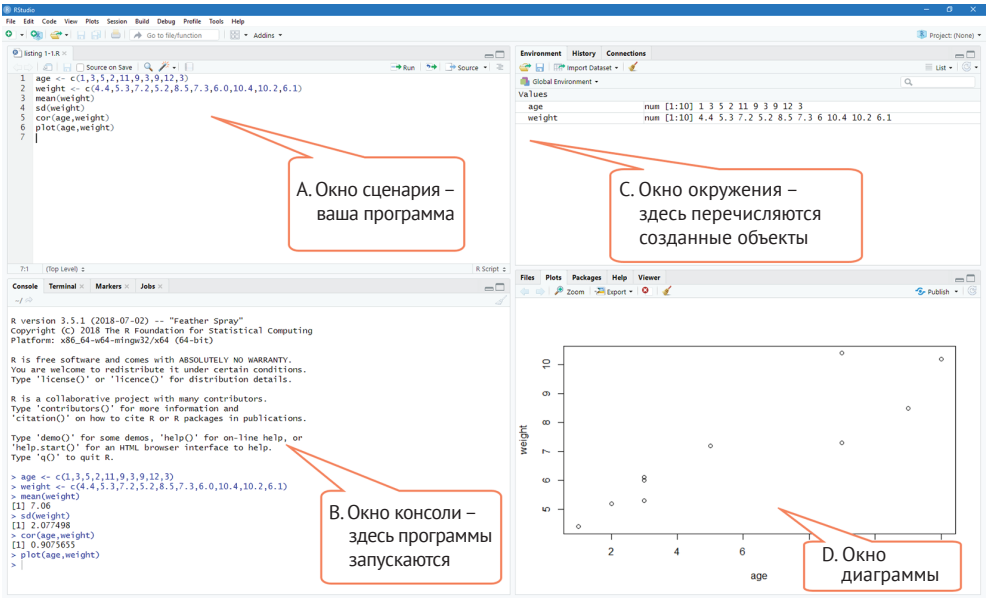

Рис. 1.5. RStudio Desktop

Окно сценария

В меню File (Файл) выберите пункт New File > R Script (Новый файл > Сценарий R). В левом верхнем углу откроется новое окно сценария (рис. 1.5 А). Введите в него код из листинга 1.1.

По мере ввода редактор будет подсвечивать синтаксис и предлагать варианты завершения кода (рис. 1.6). Например, по мере ввода `plot` появится всплывающее окно с именами доступных функций, начинающимися с букв, набранных к текущему моменту. Вы можете использовать клавиши со стрелками вверх и вниз, чтобы выбрать функцию из списка, и нажать Tab, для ее выбора. В круглых скобках, следующих за именем функции, можно нажать Tab, чтобы просмотреть список параметров функции. Нажатие Tab в кавычках приводит к завершению путей к файлам.

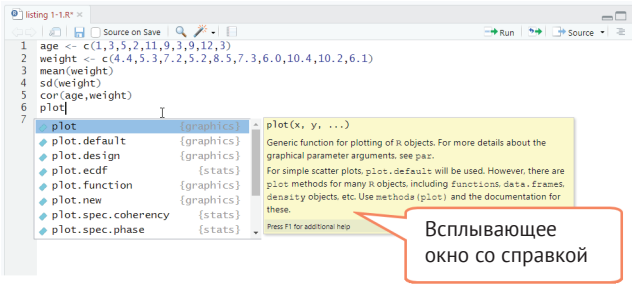

Рис. 1.6. Окно сценария

Чтобы выполнить код, выделите его и щелкните на кнопке Run (Выполнить) или нажмите Ctrl+Enter. Комбинация Ctrl+Shift+Enter запустит весь сценарий, а не только выделенный код.

Чтобы сохранить сценарий, щелкните на значке Save (Сохранить) или выберите пункт меню File > Save (Файл > Сохранить). В открывшемся диалоге выберите имя для сценария и папку, куда его следует сохранить. По соглашению файлы сценариев получают расширение `.R`. Имя файла сценария выводится на вкладке окна с красной звездочкой, если текущая версия не была сохранена.

Окно консоли

Код запускается в окне консоли (рис. 1.5 В). По сути, это та же консоль, которую можно видеть в базовом интерфейсе R. Код из окна сценария в окно консоли можно отправлять с помощью команды Run (Выполнить) или вводить команды непосредственно в этом окне, в строке приглашения к вводу (>).

Если приглашение к вводу меняется на знак плюс (+), то это означает, что интерпретатор ожидает завершения инструкции. Такое часто происходит, когда инструкция слишком длинная и не умещается на одной строке или если в коде есть непарные круглые

скобки. Вы можете отменить ввод инструкции и вернуться в командную строку, нажав клавишу Esc.

Кроме того, клавиши со стрелками вверх и вниз будут циклически переключать прошлые выполненные инструкции. Вы можете отредактировать инструкцию и повторно запустить ее клавишей Enter. Щелчок на значке с изображением метлы удаляет текст из окна.

Окна окружения и истории

Любые создаваемые объекты (в данном примере `age` и `weight`) будут появляться в окне окружения (рис. 1.5 C), а история выполненных команд будет сохраняться в окне истории – вкладка History (История) правее вкладки Environment (Окружение).

Окно диаграммы

Любые диаграммы, созданные сценарием, появятся в окне диаграммы (рис. 1.5 D). Панель инструментов этого окна позволяет циклически перемещаться по созданным диаграммам. Кроме того, окно диаграммы можно масштабировать, чтобы рассматривать диаграммы в разных масштабах, экспортировать диаграммы в несколько форматов и удалить одну или все диаграммы, созданные к настоящему моменту.

1.3.3. Как получить помощь

R имеет обширную справку. Научитесь ориентироваться в ней, и это поможет вам в работе. Встроенная система помощи содержит подробные разъяснения, ссылки на документацию и примеры для каждой функции из установленных пакетов. Справку можно вызвать при помощи функций, перечисленных в табл. 1.2.

Таблица 1.2. Функции вызова справки в R

Функция	Описание
`help.start()`	Общая справка
`help("foo")` или `?foo`	Справка по функции `foo` (кавычки необязательны)
`help(package="foo")`	Справка по пакету `foo`
`help.search("foo")` или `??foo`	Поиск в справке записей, содержащих `foo`
`example("foo")`	Примеры использования функции `foo` (кавычки необязательны)
`data()`	Список всех демонстрационных примеров данных, содержащихся в загруженных пакетах
`vignette()`	Список всех доступных руководств по загруженным пакетам
`vignette("foo")`	Список руководств по теме `foo`

Функция `help.start()` открывает окно браузера с перечнем доступных руководств разного уровня сложности, часто задаваемых

вопросов и ссылок на справочные материалы. То же самое можно получить выбором пункта меню Help > R Help (Справка > Справка по R). Функция `vignette()` вызывает список вводных статей в формате PDF или HTML. Такие статьи имеются не для всех пакетов.

Все файлы справки имеют схожий формат (рис. 1.7). Они включают заголовок и краткое описание, за которыми следует описание синтаксиса и параметров функции. Детали вычислений приводятся в разделе Details (Подробности). В разделе See Also (См. также) описаны связанные функции и ссылки на них. Страницы справки почти всегда заканчиваются примерами, иллюстрирующими типичное использование функции.

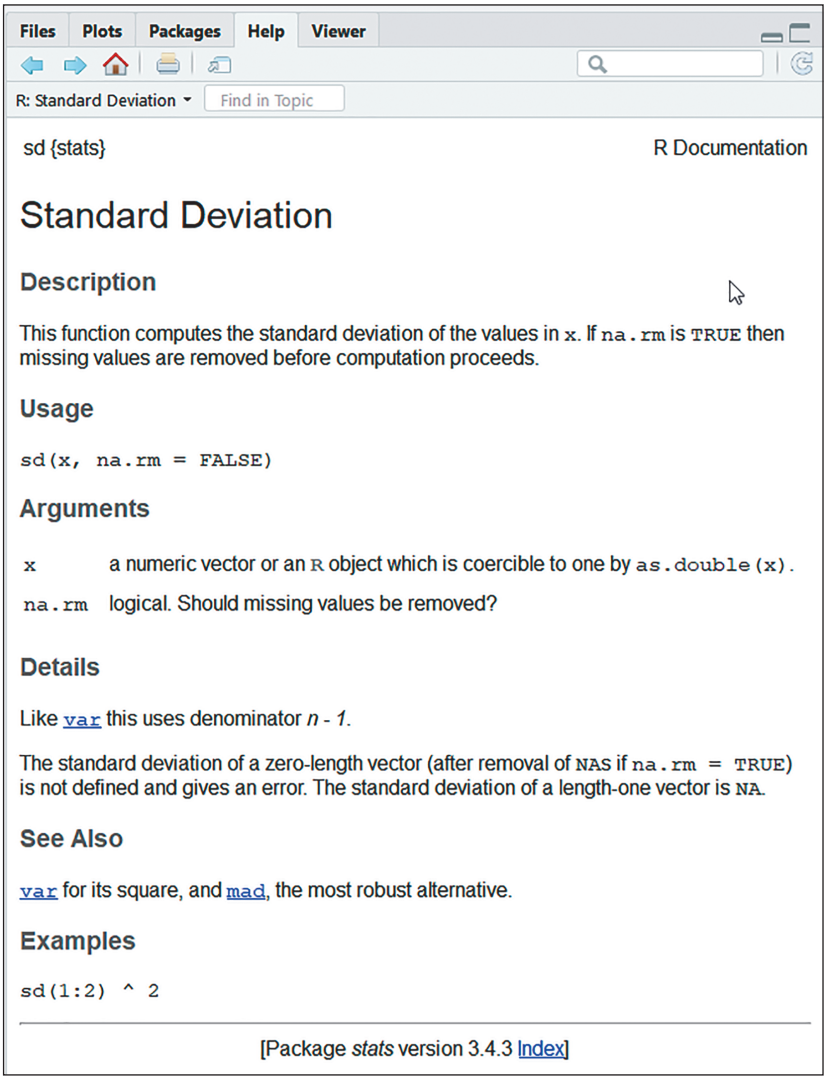

Рис. 1.7. Окно справки

Как видите, R предоставляет обширную справку, и умение ориентироваться в ней пойдет вам на пользу. Крайне редко, когда в сеансах работы с R я не использую справку, чтобы получить подробную информацию (параметры или возвращаемые значения) о какой-нибудь функции.

1.3.4. Рабочее пространство

Рабочее пространство – это текущее рабочее окружение R, включающее все созданные объекты (векторы, матрицы, функции, таблицы данных или списки). Текущий рабочий каталог – это каталог, где хранятся файлы с данными и куда по умолчанию сохраняются результаты. Узнать путь к рабочему каталогу можно с помощью функции `getwd()`. Назначить рабочий каталог можно с помощью функции `setwd()`. Чтобы импортировать файл, который находится не в рабочем каталоге, нужно указать полный путь к нему. Всегда заключайте имена файлов и каталогов в кавычки. В табл. 1.3 перечислены некоторые стандартные команды для управления рабочим пространством.

Таблица 1.3. Функции для управления рабочим пространством в R

Функция	Описание
`getwd()`	Выводит имя текущего рабочего каталога
`setwd("mydirectory")`	Назначает *`mydirectory`* текущим рабочим каталогом
`ls()`	Выводит список объектов в текущем рабочем пространстве
`rm(objectlist)`	Удаляет объекты, перечисленные в списке *`objectlist`*
`help(options)`	Выводит справку о доступных параметрах
`options()`	Позволяет просмотреть или установить текущие параметры
`save.image("myfile")`	Сохраняет рабочее пространство в файл *`myfile`* (по умолчанию `.Rdata`)
`save(objectlist, file="myfile")`	Сохраняет указанные объекты в файл *`myfile`*
`load("myfile")`	Загружает рабочее пространство в текущий сеанс

Чтобы увидеть эти функции в действии, рассмотрим следующий пример (листинг 1.2).

Листинг 1.2. Пример использования функций для управления рабочим пространством R

```
setwd("C:/myprojects/project1")
options()
options(digits=3)
```

Первая функция назначает `C:/myprojects/project1` текущим рабочим каталогом. Вторая выводит текущие значения параметров, а третья задает формат вывода чисел с тремя цифрами после запятой.

Обратите внимание, что в пути к каталогу в функции `setwd()` используются прямые слеши (/). R воспринимает обратный слеш (\) как экранирующий символ. Даже работая с R в Windows, используйте прямые слеши в путях к файлам и каталогам. Отметьте также, что функция `setwd()` не создает указанный каталог, если он не существует. Если необходимо создать каталог, используйте функцию `dir.create()`, а уже затем вызывайте `setwd()`, чтобы сделать этот каталог рабочим.

1.3.5. Проекты

Старайтесь хранить свои проекты в отдельных каталогах. RStudio предлагает для этого простой механизм. Выберите пункт меню File > New Project... (Файл > Новый проект...) и укажите либо New Directory (Новый каталог), чтобы создать проект в новом рабочем каталоге, либо Existing Directory (Существующий каталог), чтобы связать проект с существующим рабочим каталогом. Все программные файлы, история команд, отчеты, диаграммы и данные будут сохранены в каталоге проекта. Переключаться между проектами можно с помощью раскрывающегося меню Project (Проект) в верхней правой части окна RStudio.

В файлах проекта легко запутаться. Поэтому я советую создать несколько подкаталогов в основном каталоге проекта. Обычно я создаю каталог *data* для хранения файлов с исходными данными, каталог *img* для файлов изображений и создаваемых диаграмм, каталог *docs* для проектной документации и каталог *reports* для отчетов. Сценарии на R и файл *README* я храню в основном каталоге. При наличии нескольких сценариев на R, выполняемых последовательно, я нумерую их (например, *01_import_data.R*, *02_clean_data.R* и т. д.). Файл *README* – это текстовый файл, содержащий информацию об авторе, дате создания, заинтересованных сторонах и их контактах, а также цели проекта. Через полгода эта информация помогает мне вспомнить, что я сделал и зачем.

1.4. Пакеты

В базовой установке R обладает обширными возможностями. Однако некоторые наиболее впечатляющие возможности реализованы в дополнительных модулях, которые можно загрузить и установить. Существует более 10 000 созданных пользователями модулей, называемых *пакетами* (packages), которые вы можете загрузить с http://cran.r-project.org/web/packages. В них заключены почти безграничные возможности – от анализа геопространственных данных до масс-спектроскопии белков и анализа психологических тестов! В этой книге мы познакомимся со многими из них.

Особого внимания заслуживает один из наборов пакетов – `tidyverse`. Это относительно новая коллекция, предлагающая целостный и интуитивно понятный подход к обработке и анализу данных. Преимущества, предлагаемые пакетами из набора `tidyverse` (такими как `tidyr`, `dplyr`, `lubridate`, `stringr` и `ggplot2`), меняют подходы к разработке кода на R, и мы будем часто использовать эти пакеты. На самом деле необходимость описания особенностей использования этих пакетов для анализа и визуализации данных и послужила основным мотивом для выпуска этого третьего издания книги.

1.4.1. Что такое пакеты?

Пакеты – это коллекции функций на R, данных и скомпилированного программного кода в определенном формате. Каталог, где пакеты хранятся на вашем компьютере, называется *библиотекой*. Функция `.libPath()` показывает, где расположена ваша библиотека, а функция `library()` выводит названия всех имеющихся в библиотеке пакетов.

В дистрибутив R уже входит стандартный набор пакетов (включая `base`, `datasets`, `utils`, `gr-Devices`, `graphics`, `stats` и `methods`). В них уже содержатся разнообразные функции и наборы данных, доступные по умолчанию. Также есть возможность скачивать и устанавливать дополнительные пакеты. После установки они загружаются в ходе сеанса по мере необходимости. Функция `search()` выводит названия загруженных и готовых к использованию пакетов.

1.4.2. Установка пакета

В R существует множество функций для управления пакетами. Установить пакет можно с помощью функции `install.packages()`. Например, есть такой пакет, как `gclus`, который содержит функции для создания улучшенных диаграмм рассеяния. Этот пакет можно скачать и установить вызовом функции `install.packages("gclus")`.

Пакет нужно установить только один раз. Однако, как и любые другие программы, пакеты часто обновляются их разработчиками. Обновить все установленные пакеты можно вызовом функции `update.package()`. Для получения информации об установленных пакетах используйте функцию `installed.packages()`. Она выведет список всех установленных пакетов с номерами их версий, названиями пакетов, от которых они зависят, и другой информацией.

Устанавливать и обновлять пакеты можно также из интерфейса RStudio. Выберите вкладку Packages (Пакеты) в окне справа внизу. Введите имя (или часть имени) в поле поиска в правом верхнем углу этого окна со вкладками. Поставьте галочки напротив пакетов, которые вы хотите установить, и щелкните на кнопке Install (Установить) или Update (Обновить), чтобы обновить уже установленные пакеты.

1.4.3. Загрузка пакета

В процессе установки пакет сначала скачивается с сайта CRAN в вашу библиотеку. Для использования этого пакета в текущем сеансе нужно загрузить его вызовом функции `library()`. Например, чтобы использовать пакет `gclus`, введите команду `library(gclus)`.

Разумеется, прежде чем загрузить пакет, его необходимо установить. В течение сеанса достаточно загрузить пакет только один раз. При необходимости можно настроить рабочее пространство так, чтобы часто используемые пакеты загружались автоматически в начале каждого сеанса. Настройка рабочего окружения подробно описана в приложении B.

1.4.4. Получение информации о пакете

После загрузки пакета становятся доступны новые функции и наборы данных. Небольшие наборы данных поставляются вместе с демонстрационным программным кодом, что позволяет протестировать новые возможности. Справочная система содержит описание каждой функции (с примерами) и информацию о каждом встроенном наборе данных. Функция `help(package="имя_пакета")` выведет краткое описание указанного пакета и список всех входящих в него функций и наборов данных. Вызов `help()` с именами этих функций или наборов данных позволит выяснить новые детали. Эту информацию можно также найти на сайте CRAN в виде руководства в формате PDF.

Чтобы получить справку по пакету в интерфейсе RStudio, щелкните на вкладке Packages (Пакеты) в окне внизу справа, введите имя пакета в окне поиска и щелкните на имени пакета.

Распространенные ошибки в программировании на R

Существует ряд распространенных ошибок, которые часто допускают и новички, и опытные программисты. Если программа выдает сообщение об ошибке, проверьте, не сделали ли вы что-то из нижеперечисленного:

- *использовали неправильный регистр:* `help()`, `Help()` *и* `HELP()` *– это три разные функции (только первое имя правильное);*
- *забыли поставить кавычки там, где они необходимы:* `install.packages(«gclus»)` *работает, а* `install.packages(gclus)` *выдаст сообщение об ошибке;*
- *забыли добавить круглые скобки в вызов функции: например,* `help()` *– правильный вызов, а* `help` *– нет. Скобки должны добавляться, даже когда функция вызывается без аргументов;*
- *использовали* `\` *в пути к файлу в операционной системе Windows: R воспринимает обратный слеш как экранирующий символ. Вызов* `setwd(«c:\mydata»)` *сгенерирует сообщение об ошибке. Используйте* `setwd(«c:/mydata»)` *или* `setwd(«c:\\mydata»)`*;*

- *использовали функцию из пакета, который еще не загрузили. В пакете* `gclus` *имеется функция* `order.clusters()`. *Если попытаться вызвать ее до загрузки пакета* `gclus`, *то появится сообщение об ошибке.*

Сообщения об ошибках в R могут быть непонятными, однако появление многих из них можно предотвратить, если внимательно следовать вышеперечисленным правилам.

1.5. Передача вывода на ввод: повторное использование результатов

Одна из наиболее полезных особенностей R – возможность сохранить результаты анализа и использовать в качестве входных данных в дополнительном анализе. Рассмотрим пример, воспользовавшись одним из наборов данных, распространяемых вместе с R. Если какие-то детали будут вам непонятны, не волнуйтесь. Здесь важно понять общий принцип, а не частности.

Вместе с R распространяется множество встроенных наборов данных, на которых можно попрактиковаться в анализе данных. Один из таких наборов, с именем `mtcars`, содержит информацию о 32 автомобилях, собранную в ходе дорожных испытаний журналом «Motor Trend». Предположим, нам нужно описать взаимосвязь между топливной эффективностью автомобиля и его весом.

Для начала можно попробовать выполнить простую линейную регрессию, предсказывающую, сколько миль проедет на одном галлоне[1] топлива (`mpg`) автомобиль с заданным весом (`wt`):

```
lm(mpg~wt, data=mtcars)
```

Результаты появятся на экране, но не будут сохранены.

Ту же регрессию можно вычислить с сохранением результатов в объекте:

```
lmfit <- lm(mpg~wt, data=mtcars)
```

Операция присваивания создаст объект списка с именем `lmfit` и сохранит в нем обширную информацию с результатами анализа (включая прогнозируемые значения, остатки, коэффициенты регрессии и т. д.). В этом случае на экран ничего не выводится, но результаты сохраняются, и их можно вывести на экран или подвергнуть дальнейшей обработке.

Вызов `summary(lmfit)` отобразит сводку результатов, а вызов `plot(lmfit)` выведет прогностические графики. Инструкция `cook<-cooks.distance(lmfit)` сгенерирует и сохранит затронутые статистики, а `plot(cook)` выведет ее график. Чтобы спрогнозировать расстоя-

[1] 1 галлон ≈ 4,5 л в Англии и ≈ 3,7 л в США. – *Прим. перев.*

ние в милях, которое проедет на одном галлоне автомобиль с некоторым весом, нужно выполнить вызов `predict(lmfit, mynewdata)`.

Чтобы узнать, что возвращает заинтересовавшая вас функция, загляните в раздел Value (Значение) на странице справки для этой функции. Например, вызвав `help(lm)` или `?lm`, вы узнаете, что сохранит операция присваивания результатов этой функции объекту.

1.6. Работа с большими массивами данных

Программисты часто спрашивают меня, может ли R обрабатывать большие массивы данных. Как правило, они работают со значительными объемами данных, собранных в процессе исследований климата или генетики. R хранит объекты в памяти, поэтому объем ОЗУ является ограничивающим фактором. Например, на моем девятилетнем компьютере с операционной системой Windows и 2 Гбайт оперативной памяти я могу обрабатывать наборы данных с 10 млн элементов (100 признаков 100 000 объектов). На iMac с 4 Гбайт оперативной памяти без особых затруднений можно обрабатывать наборы данных, включающие 100 млн элементов.

Но есть два аспекта, которые следует учитывать: размер набора данных и применяемые статистические методы. R способен обрабатывать наборы данных в диапазоне от гигабайта до терабайта, но для этого необходимо применять специальные процедуры. Подробнее об анализе больших массивов данных рассказывается в приложении F.

1.7. Учимся на примере

Закончим эту главу примером, объединяющим многие рассмотренные идеи. Вот задание:

1. Откройте общий файл справки и загляните в раздел Introduction to R (Введение в R).
2. Установите пакет `vcd` (пакет для визуализации категориальных данных, который мы подробно рассмотрим в главе 11).
3. Выведите список всех функций и наборов данных в этом пакете.
4. Загрузите пакет в текущий сеанс и прочтите описание набора данных `Arthritis`.
5. Выведите этот набор данных на экран (набрав его имя в командной строке).
6. Запустите пример, прилагаемый к набору данных `Arthritis`. Не волнуйтесь, если результаты покажутся вам непонятными. Если в двух словах, то они показывают, что страдающие артритом пациенты, получавшие лекарство, выздоравливали гораздо быстрее, чем получавшие плацебо.

В листинге 1.3 приводится программный код, который вам понадобится, а некоторые результаты приведены на рис. 1.8. Как показывает это упражнение, с помощью короткого фрагмента программного кода можно сделать многое.

Листинг 1.3. Работа с новым пакетом

```
help.start()
install.packages("vcd")
help(package="vcd")
library(vcd)
help(Arthritis)
Arthritis
example(Arthritis)
```

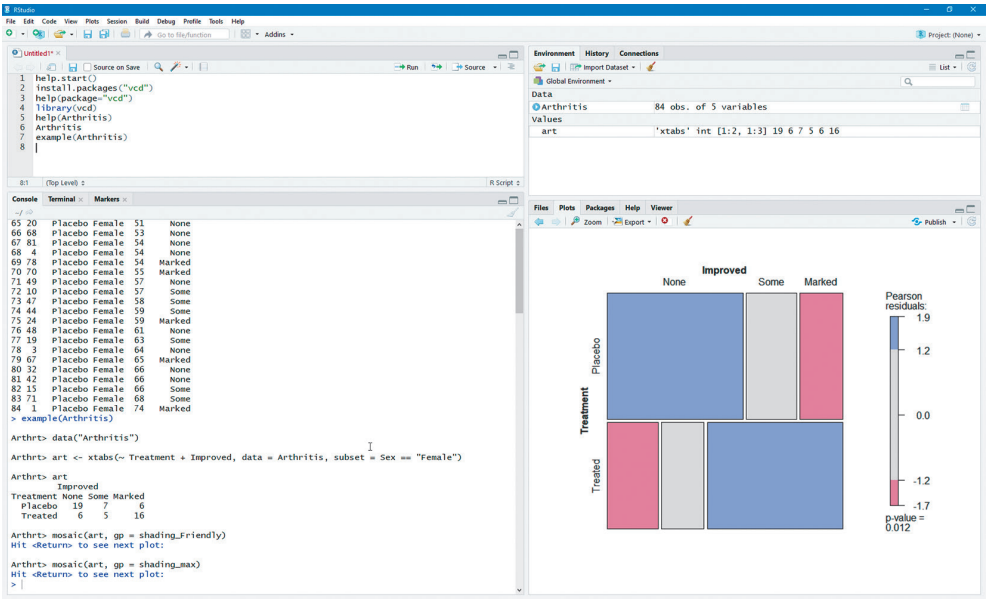

Рис. 1.8. Окно RStudio после выполнения кода в листинге 1.3

В этой главе мы рассмотрели некоторые сильные стороны R, особенно привлекательные для студентов, исследователей, статистиков и аналитиков, желающих анализировать свои данные. Мы рассмотрели порядок установки платформы R и расширения ее возможностей с использованием дополнительных пакетов. Исследовали основные характеристики интерфейса и создали несколько графиков для примера. R – сложная платформа, поэтому мы потратили некоторое время на знакомство с ее справочной системой. Надеюсь, вы уже почувствовали, насколько мощным может быть бесплатное программное обеспечение.

Теперь, установив R и RStudio, можно приступать к анализу данных. В следующей главе мы познакомимся с типами данных, поддерживаемыми в R, и узнаем, как импортировать данные из текстовых файлов, других программ и систем управления базами данных.

Итоги

- R предоставляет мощную интерактивную среду для анализа и визуализации данных.
- RStudio – это интегрированная среда разработки, делающая программирование на R проще и продуктивнее.
- Пакеты – это бесплатные дополнительные модули, расширяющие возможности платформы R.
- R имеет обширную справочную систему, и умение ее использовать значительно облегчит программирование и повысит его эффективность.