

АНАЛИЗ ВРЕМЕННЫХ РЯДОВ С ПОМОЩЬЮ R



Мастицкий С. Э.

Анализ временных рядов с помощью R

Мастицкий С. Э.

2020

Мастицкий С. Э. (2020) Анализ временных рядов с помощью R. — Электронная книга, адрес доступа: <https://ranalytics.github.io/tsa-with-r>

Эта книга представляет собой небольшое пособие по использованию языка программирования и системы статистических вычислений R для анализа временных рядов. Упор сделан на решение нескольких стандартных задач, включая прогнозирование, выявление структурных изменений и аномалий в данных, а также кластеризацию временных рядов. Описание соответствующих подходов и программного обеспечения сопровождается многочисленными примерами кода в применении к данным из реального мира. Книга рассчитана на опытных пользователей R, которым знакомы принципы построения предсказательных моделей, ряд стандартных методов статистики (регрессия, метод главных компонент, кластерный анализ), а также основы байесовской статистики.

Данная работа распространяется в рамках лицензии Creative Commons “Атрибуция — Некоммерческое использование — На тех же условиях 4.0 Всемирная” (CC BY-NC-SA 4.0). Согласно этой лицензии, Вы можете свободно копировать, распространять и видоизменять данное произведение при условии точного указания его автора и источника. При изменении этого произведения или использовании его в своих работах, Вы можете распространять результат только по такой же или подобной лицензии. Запрещается использовать эту работу в коммерческих целях без согласования с автором. Более подробная информация о лицензии представлена на странице <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.ru>



© 2020, Сергей Эдуардович Мастицкий
Эл. почта: rtutorialsbook@gmail.com
Веб-сайт: <http://r-analytics.blogspot.com>

*С благодарностью ко всем читателям моего блога.
Без вашей поддержки эта книга не появилась бы на свет.*

Сергей М., апрель 2020 г., Лондон

Оглавление

1	Введение	6
1.1	О чем эта книга и чего в ней нет	6
1.2	Что ожидается от читателя	7
1.3	Основные понятия	7
1.4	Формат данных tsibble	8
1.5	Данные, используемые в примерах	11
I	Разведочный анализ данных	17
2	Обработка пропущенных наблюдений	18
3	Агрегирование наблюдений	23
4	Визуализация временных рядов и их свойств	27
5	Извлечение признаков с помощью пакета feasts	36
5.1	Функция features()	36
5.2	Встроенные функции для расчета признаков	38
5.3	Примеры использования извлеченных признаков	44
II	Прогнозирование временных рядов	48
6	Пакет prophet	49
6.1	Методология	50
6.2	Первый простой пример	50
6.3	Функция prophet()	53
6.4	Точки излома тренда	56
6.5	Эффекты праздников и других важных событий	60
6.6	Сезонные компоненты	67
6.7	Модели с предикторами	75
6.8	Выбор оптимальной модели	81
6.9	Моделирование емкости системы	89
7	Пакет bsts	95
7.1	Методология	95
7.2	Функция bsts()	97

7.3	Спецификация компонент модели	99
7.4	Примеры моделей без предикторов	101
7.5	Модели с предикторами	110
7.6	Выбор оптимальной модели	117

III Структурные изменения и аномалии во временных рядах 121

8	Выявление структурных изменений	122
8.1	Метод “E-Divisive with Medians” (EDM)	122
8.2	Функция breakout()	123
8.3	Примеры использования функции breakout()	124
9	Выявление аномалий	129
9.1	Автоматическое обнаружение аномалий	129
9.2	Ручная настройка параметров для обнаружения аномалий	130
9.3	Одновременный анализ нескольких временных рядов	136

IV Кластеризация 139

10	Задача кластеризации временных рядов	140
11	Кластеризация по исходным данным	142
11.1	DTW-расстояние	142
11.2	Пакет dtwclust	147
11.3	Пример кластеризации с использованием DTW-расстояния	149
12	Кластеризация по описательным признакам	157
13	Кластеризация по результатам подгонки моделей	162

Глава 1

Введение

1.1. О чем эта книга и чего в ней нет

Несколько лет назад, когда я впервые столкнулся с задачей по прогнозированию выручки компании, в которой на тот момент работал, я не знал с чего начать, потому что прежде не имел дела с временными рядами. Как это часто бывает, я обратился к [CRAN Task Views](#), чтобы получить общее представление о том, какие пакеты для R подошли бы для решения моей задачи. Увиденное разнообразие инструментов и большой объем знаний, которые потребовались бы для уверенного пользования этими инструментами, вызвали у меня небольшую панику. С чего начать? Какой пакет выбрать, чтобы потратить минимальное время на его изучение и успешно завершить проект в срок? Какую обзорную статью почитать, чтобы быстро вникнуть в тот или иной метод?

Эта книга — моя скромная попытка помочь тем, кто оказался в похожей ситуации. Анализ временных рядов — обширная тема, по которой есть очень много литературы, и у меня ни в коей мере не было цели написать еще одну теоретическую работу. Вместо этого я делаю упор на решение нескольких стандартных задач и делюсь с читателями многочисленными примерами кода в применении к данным из реального мира. Описанное здесь — это методы и программное обеспечение, которые я добавил в свой личный “набор инструментов” в результате работы над несколькими аналитическими проектами и которые показали свою практическую ценность. Надеюсь, что эти инструменты окажутся полезными и для читателей. Теоретическая часть обсуждаемых методов представлена лишь в том объеме, который необходим для осознанного использования соответствующих пакетов для R. Дополнительные теоретические выкладки можно найти в приведенных в тексте ссылках на ключевые публикации.

Большинство представленных здесь материалов ранее было опубликовано в моем блоге “[R: Анализ и визуализация данных](#)”. Не затронутые в блоге, но добавленные в книгу темы включают визуализацию различных свойств временных рядов (гл. 4), прогнозирование с помощью байесовских структурных моделей (гл. 7) и кластерный анализ временных рядов (гл. 10–13).

1.2. Что ожидается от читателя

Книга в основном рассчитана на опытных пользователей языка программирования и системы статистических вычислений R. Поэтому ожидается, что читателю знакомы:

- основные пакеты из группы tidyverse (в частности, dplyr, ggplot2, tibble, и tidyr);
- принципы построения предсказательных моделей;
- такие стандартные методы статистики и машинного обучения, как кусочно-линейная регрессия, обобщенные аддитивные модели, метод главных компонент и кластерный анализ;
- основы байесовской статистики.

Если вы только начинаете работать с R, то я порекомендовал бы сначала обратиться к вводным главам любой из следующих книг на русском языке:

- Кабаков Р.И. *Анализ и визуализация данных в программе R* / пер. с англ. П. Волковой. М.: ДМК Пресс, 2014.
- Мастицкий С.Э., Шитиков В.К. *Статистический анализ и визуализация данных с помощью R*. М.: ДМК Пресс, 2015.
- Уикем Х., Гроулмунд Г. *Язык R в задачах науки о данных. Импорт, подготовка, обработка, визуализация и моделирование данных* / пер. с англ. А.Г. Гузикевича. М.: Вильямс, 2018.

Приведенные в этой книге примеры можно без труда воспроизвести на любом современном компьютере. Для этого лишь потребуются R с версией не ниже 3.6.3 и любая удобная для вас интегрированная среда разработки (например, [RStudio](#)). Большинство пакетов для R, используемых в примерах, можно установить стандартным образом из хранилища CRAN с помощью функции `install.packages()`. В редких случаях, когда тот или иной пакет отсутствует в [CRAN](#), я привожу дополнительные инструкции по его установке. Небольшой скрипт для инсталляции всех нужных пакетов есть в Github-репозитории [ranalytics/tsa-r](#). Используйте, пожалуйста, раздел Issues этого репозитория также для сообщения об обнаруженных в тексте ошибках и неточностях.

1.3. Основные понятия

Временной ряд представляет собой последовательность значений некоторой переменной (или переменных), регистрируемых через определенные промежутки времени (регулярные или нерегулярные). Когда есть только одна наблюдаемая переменная, временной ряд называют *одномерным*. В случае же с несколькими параллельно наблюдаемыми переменными говорят о *многомерном* временном ряде. Мы будем рассматривать только одномерные ряды.

Временные ряды можно встретить в самых разных областях — от метеорологии, где ведется учет данных по погодным показателям, до [Интернета вещей](#), где регистрируются сигналы от всевозможных встроенных датчиков и сенсоров. Под *анализом временных рядов* мы будем понимать процесс применения методов статистики и машинного обучения для выявления закономерностей в структуре временных рядов и предсказания будущего поведения описываемых этими рядами систем. В частности, будут рассмотрены следующие распространенные задачи:

- *прогнозирование*, т.е. предсказание будущих значений временного ряда;
- выявление *структурных изменений* и *аномалий*, вызванных в изучаемой системе влиянием внешних или внутренних факторов (например, изменения в экономических показателях, связанные с политическими событиями в стране; всплески уровней продаж, обусловленные рекламными кампаниями; кратковременные неисправности в технических системах, и т.п.);
- *кластеризация*, т.е. нахождение групп временных рядов, похожих по своим свойствам.

В общем случае одномерный временной ряд из наблюдений y_t , учтенных в моменты времени t , можно разложить на следующие составляющие, или *компоненты*:

- *тренд* (T_t): характеризует долговременную тенденцию в данных (снижение или возрастание). Тренд может быть линейным или нелинейным. В некоторых временных рядах может также наблюдаться изменение направления тренда (например, когда рост сменяется спадом).
- *циклическая компонента* (C_t): долговременные циклические колебания, обычно занимающие не менее 2 лет. Как правило, *частота* таких изменений непостоянна (например, хорошо известен *11-летний цикл* солнечной активности, хотя на самом деле длительность этого цикла варьирует).
- *сезонная компонента* (S_t): кратковременные периодические изменения, обладающие фиксированной частотой (например, суточные изменения количества солнечного света, падающего на единицу поверхности Земли).
- *нерегулярная компонента* (ϵ_t): эффекты случайных факторов (“шум”).

Хотя функциональная связь между перечисленными компонентами может принимать практически любую форму, обычно рассматривают зависимости следующих двух видов:

- *аддитивная модель*: $y_t = T_t + C_t + S_t + \epsilon_t$;
- *мультипликативная модель*: $y_t = T_t \times C_t \times S_t \times \epsilon_t$.

Часто длина временного ряда оказывается недостаточной для выделения циклической составляющей, в связи чем C_t исключают из рассмотрения.

Аддитивную модель применяют к т.н. *стационарным* временным рядам, в которых среднее значение и дисперсия y_t примерно постоянны для всех t . Мультипликативная же модель лучше подходит для описания *нестационарных* рядов, в которых обычно имеют место выраженный тренд и возрастание дисперсии y_t во времени.

На рис. 1.1 приведены примеры временных рядов, свойства которых иллюстрируют описанные выше понятия. Дополнительные понятия и термины мы будем вводить по мере необходимости в ходе рассмотрения соответствующих методов анализа.

1.4. Формат данных tsibble

Анализ временных рядов в R часто сопровождается приличной “головной болью”, что связано с необходимостью представления данных в виде объектов таких традиционных классов, как `ts`, `zoo` или `xts`. К сожалению, эти классы плохо

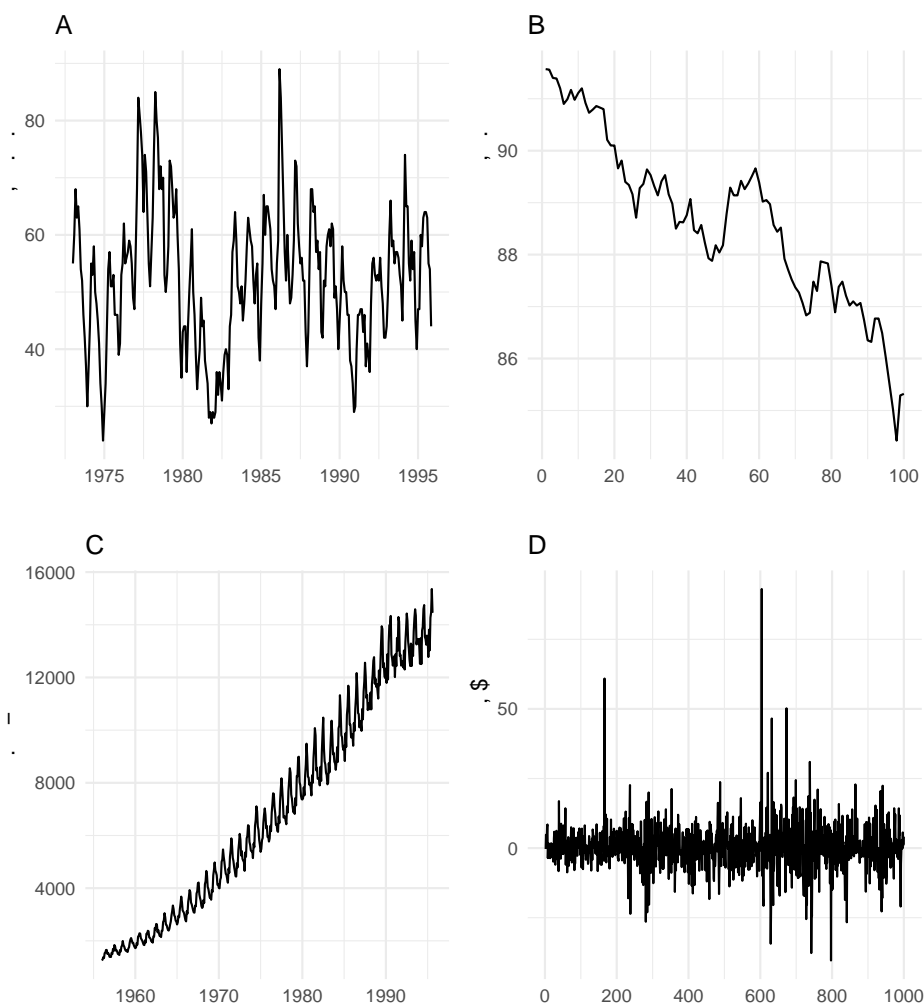


РИСУНОК 1.1. (А): Месячные продажи новых домов в США демонстрируют заметную годовую сезонность, а также более долговременную цикличность с периодом в 6–10 лет. В этих данных нет выраженного тренда. (В): Количество фьючерных контрактов Департамента казначейства США, проданных на Чикагской фондовой бирже в течение 100 дней в 1981 г. В этом временном ряду отсутствуют сезонные колебания, но имеет место тренд на снижение. (С): Квартальные объемы выработки электроэнергии в Австралии демонстрируют четко выраженные тренд и сезонность. Кроме того, дисперсия данных возрастает во времени, что указывает на мультипликативность породившего их процесса. (D): В дневных изменениях цены акций Google на момент закрытия торгов нет ни тренда, ни сезонности, ни цикличности. *Источник:* Hyndman and Athanasopoulos (2019)

подходят для современных наборов данных, которые часто характеризуются нерегулярной регистрацией наблюдений во времени, наличием нескольких переменных разных типов, нескольких группирующих переменных и т.п. Кроме того, традиционные форматы представления временных рядов противоречат принципам организации и хранения “опрятных данных” (“tidy data”) и, как следствие, затрудняют анализ и моделирование с помощью широко используемых сегодня инструментов из группы `tidyverse`. Для решения этих проблем группа исследователей под руководством проф. Роба Хиндмана (Rob Hyndman) разработала новый формат, реализованный в пакете `tsibble` (Wang et al., 2020). Именно этот формат мы будем использовать для представления данных в большинстве рассматриваемых ниже примеров. Он характеризуется следующими свойствами:

- данные хранятся в табличном виде;
- в таблице должны присутствовать как минимум два столбца — со значениями наблюдаемой во времени количественной переменной и с упорядоченными по возрастанию (т.е. от прошлого к будущему) временными отметками (столбец с временными отметками называется *индексирующим* — `index`);
- кроме того, в таблицу могут входить одна или несколько *группирующих* переменных (`key`) — значения этих переменных указывают на принадлежность каждого наблюдения к соответствующему временному ряду;
- любое наблюдение в таблице можно уникально идентифицировать по сочетанию значений индексирующей и группирующих переменных.

Для создания объектов класса `tsibble` служит функция `as_tsibble()`, которая имеет следующие аргументы:

- `x` — объект с данными, подлежащий преобразованию в объект класса `tsibble` (это может быть, например, числовой вектор, матрица, таблица с данными (`data.frame` или `tibble`) и др.).
- `index` — переменная с временными отметками (указывается без кавычек). Допускается использование временных отметок на шкале от наносекунд до года.
- `key` — одна или несколько группирующих, или ключевых, переменных, которые уникально определяют каждый хранящийся в таблице временной ряд. Названия переменных указываются без кавычек и объединяются с помощью функции конкатенации `c()`. По умолчанию этот аргумент равен `NULL`, т.е. предполагается, что группирующих переменных в таблице нет.
- `regular` — логический аргумент, который указывает на регулярность учета хранящихся в таблице наблюдений. Значение `TRUE` (принято по умолчанию) предполагает, что учет выполнялся с одинаковым интервалом (например, каждую минуту, час, день, и т.п.).
- `validate` — логический аргумент (по умолчанию равен `TRUE`), позволяющий выполнить проверку уникальности каждого наблюдения по сочетанию значений переменных `index` и `key`. Если вы уверены, что каждое наблюдение уникально, то такую проверку можно отключить (`FALSE`) — это приведет к более быстрому выполнению команды `as_tsibble()` в случае с большими наборами данных.
- `.drop` — логический аргумент, позволяющий исключить из таблицы “пустые” временные ряды, т.е. такие сочетания значений группирующих переменных, для которых значения `x` отсутствуют.

Примеры работы с функцией `as_tsibble()` приведены в следующем разделе.

1.5. Данные, используемые в примерах

Для воспроизведения рассматриваемых в этой книге примеров необходимо скопировать содержимое Github-репозитория [ranalytics/tsa-r](#) любым удобным для вас способом и сделать корневую директорию скопированного проекта рабочей директорией R. Используемые в примерах четыре набора данных хранятся в папке `data` этого проекта. Ниже приведено их краткое описание.

1.5.1. Стоимость 22 криптовалют

Набор данных `cryptos` содержит собранные с сайта [CoinMarketCap](#) значения стоимости 22 [криптовалют](#) на момент закрытия торгов. Эти данные охватывают период с 1 января 2018 г. по 6 декабря 2019 г. и имеют очень простую структуру:

```
require(readr)
require(tibble)

cryptos <- read_csv("data/cryptos_price.csv")
glimpse(cryptos, width = 60)
```

```
## Observations: 15,510
## Variables: 3
## $ y      <dbl> 7547.00, 7448.31, 7252.03, 7320.15, 7321.9...
## $ ds     <date> 2019-12-06, 2019-12-05, 2019-12-04, 2019-...
## $ coin   <chr> "bitcoin", "bitcoin", "bitcoin", "bitcoin"...
```

Переменная `y` — это стоимость криптовалюты `coin` (в долларах США), отмеченная в день `ds`. Все 22 временных ряда изображены на рис. 1.2.

```
require(dplyr)
require(ggplot2)
require(ggplot2)

cryptos %>%
  group_by(coin) %>%
  mutate(label = ifelse(ds == max(ds), coin, NA)) %>%
  ggplot(., aes(ds, y, group = coin)) +
  geom_line() +
  geom_text_repel(aes(label = label),
    size = 3, nudge_x = 50,
    segment.size = 0.4,
    segment.color = "gray60",
    point.padding = 0.2,
    force = 5, na.rm = TRUE) +
  scale_y_log10() +
  theme_minimal() +
  xlim(c(as.Date("2018-01-01"), as.Date("2020-06-01")))
```

Заметьте, что стоимость одной из криптовалют (`tether`) на протяжении всего исследованного периода была настолько низковолатильной, что ее временной ряд на рис. 1.2 выглядит почти как сплошная горизонтальная линия.

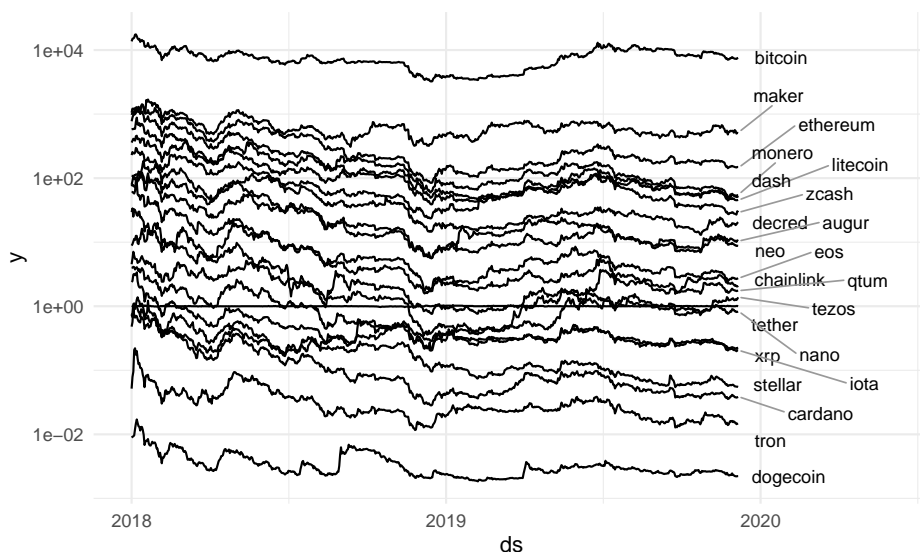


РИСУНОК 1.2. Динамика стоимости 22 криптовалют. *Источник:* [CoinMarketCap](#)

Воспользуемся функцией `as_tibble()` для преобразования таблицы `cryptos` в объект класса `tsibble`:

```
require(tsibble)

(cryptos <- as_tibble(cryptos, key = coin, index = ds))
```

```
## # A tsibble: 15,510 x 3 [1D]
## # Key:      coin [22]
##       y ds      coin
##   <dbl> <date>   <chr>
## 1  74.5 2018-01-01 augur
## 2  79.5 2018-01-02 augur
## 3  77.5 2018-01-03 augur
## 4  73.7 2018-01-04 augur
## 5  72.8 2018-01-05 augur
## 6  77.5 2018-01-06 augur
## 7  80.2 2018-01-07 augur
## 8  98.1 2018-01-08 augur
## 9  91.9 2018-01-09 augur
## 10 103. 2018-01-10 augur
## # ... with 15,500 more rows
```

Если внимательно присмотреться, то можно увидеть некоторые внешние отличия полученной таблицы от исходной `cryptos`:

- R теперь “знает”, что таблица `cryptos` является объектом класса `tsibble` с 15510 наблюдениями, учтенными с дневным интервалом (см. комментарий # A tsibble: 15,510 x 3 [1D]).

- у таблицы `cryptos` есть группирующая переменная `coin` с 22 уровнями (см. `# Key: coin [22]`). Все наблюдения каждого временного ряда упорядочены по возрастанию значений индексирующей переменной `ds`.

Однако это всего лишь внешние различия. Гораздо важнее то, что к данным из таблицы `cryptos` теперь можно применять методы анализа, специфичные для временных рядов, и делать это обычным для инструментов `tidyverse` образом. Аналогичные преобразования в формат `tsibble` будут выполнены нами ниже и для других наборов данных, используемых в примерах из этой книги.

1.5.2. Стоимость биткоина

Набор данных `bitcoin` содержит собранные с сайта [CoinMarketCap](#) значения стоимости биткоина на момент закрытия торгов в период с 1 января 2016 г. по 24 августа 2019 г.

```
bitcoin <- read_csv("data/bitcoin_price.csv") %>%
  as_tsibble(., key = NULL, index = ds)
glimpse(bitcoin, width = 60)
```

```
## Observations: 1,332
## Variables: 2
## $ y <dbl> 434.33, 433.44, 430.01, 433.09, 431.96, 429....
## $ ds <date> 2016-01-01, 2016-01-02, 2016-01-03, 2016-01...
```

Переменная `y` — это стоимость биткоина (в долларах США), отмеченная в день `ds`. Заметьте, что в отличие от рассмотренного выше набора данных `cryptos`, таблица `bitcoin` содержит только один временной ряд. Поэтому при преобразовании этой таблицы в формат `tsibble` аргументу `key` было присвоено значение `NULL` — таким образом мы сообщили программе, что в таблице нет группирующих переменных. Динамика стоимости биткоина в рассматриваемый период времени изображена на рис. 1.3.

```
bitcoin %>%
  ggplot(., aes(ds, y)) +
  geom_line() + scale_y_log10() +
  theme_minimal()
```

1.5.3. Цена акций трех компаний

Набор данных `shares` содержит значения цены акций компаний Amazon, Facebook и Google на момент закрытия торгов в период с 1 января 2016 г. по 26 мая 2019 г. Эти данные были собраны с сайта [Yahoo Finance](#) и по своей структуре напоминают описанную выше таблицу `cryptos`:

```
shares <- read_csv("data/shares_price.csv") %>%
  as_tsibble(., key = share, index = ds)
glimpse(shares, width = 60)
```



РИСУНОК 1.3. Динамика стоимости биткоина. *Источник:* [CoinMarketCap](#)

```
## Observations: 3,726
## Variables: 3
## Key: share [3]
## $ ds      <date> 2016-01-01, 2016-01-02, 2016-01-03, 2016...
## $ share <chr> "amzn", "amzn", "amzn", "amzn", "amzn", "...
## $ price <dbl> NA, NA, NA, 636.99, 633.79, 632.65, 607.9...
```

Переменная `price` — это цена (в долларах США) акции `share`, отмеченная в день `ds`. Обратите внимание на наличие пропущенных значений в переменной `price`, связанное с прекращением торгов в выходные дни и во время государственных праздников США. Эти пропущенные значения соответствуют многочисленным “пробелам” во временных рядах, изображенных на рис. 1.4.

```
shares %>%
  ggplot(., aes(ds, price)) +
  geom_line() + scale_y_log10() +
  facet_wrap(~share, ncol = 1, scales = "free_y") +
  theme_minimal()
```

1.5.4. Стоимость номеров в трех гостиницах

Набор данных `hotels` содержит значения суточной стоимости номеров в трех гостиницах за период с 1 ноября 2012 г. по 30 июня 2013 г. Эта таблица является частью гораздо большего набора данных, предоставленного компанией Expedia в рамках одного из [Kaggle-соревнований](#). В таблицу входят три переменные: `prop_id` (идентификатор гостиницы), `date_time` (время, когда пользователь увидел цену на соответствующий гостиничный номер в результате онлайн-поиска) и `price_usd` (цена в долларах США):

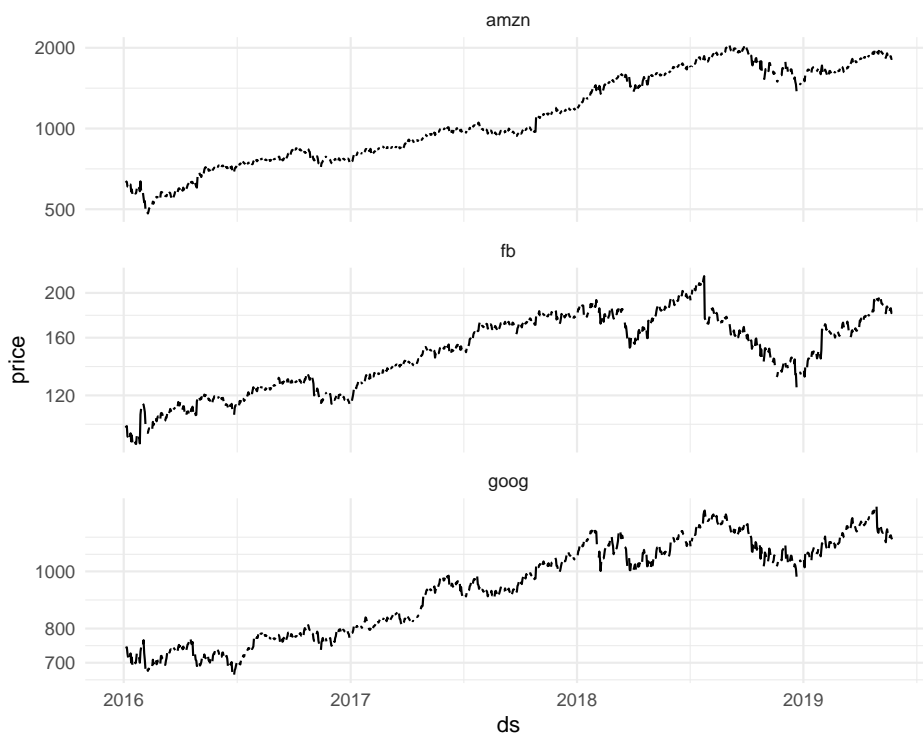


РИСУНОК 1.4. Динамика цены акций компаний Amazon, Facebook и Google.
Источник: [Yahoo Finance](#)

```
hotels <- read_csv("data/hotels_price.csv") %>%  
  as_tsibble(., key = prop_id, index = date_time)  
glimpse(hotels, width = 60)
```

```
## Observations: 1,647  
## Variables: 3  
## Key: prop_id [3]  
## $ prop_id   <dbl> 13252, 13252, 13252, 13252, 13252, 13...  
## $ date_time <dtm> 2012-11-01 10:16:16, 2012-11-02 17:5...  
## $ price_usd <dbl> 184.00, 203.00, 203.00, 186.00, 169.0...
```

Динамика стоимости номеров в рассматриваемых трех гостиницах изображена на рис. 1.5.

```
hotels %>%  
  ggplot(., aes(date_time, price_usd)) +  
  geom_line() + facet_wrap(~prop_id) +  
  theme_minimal()
```

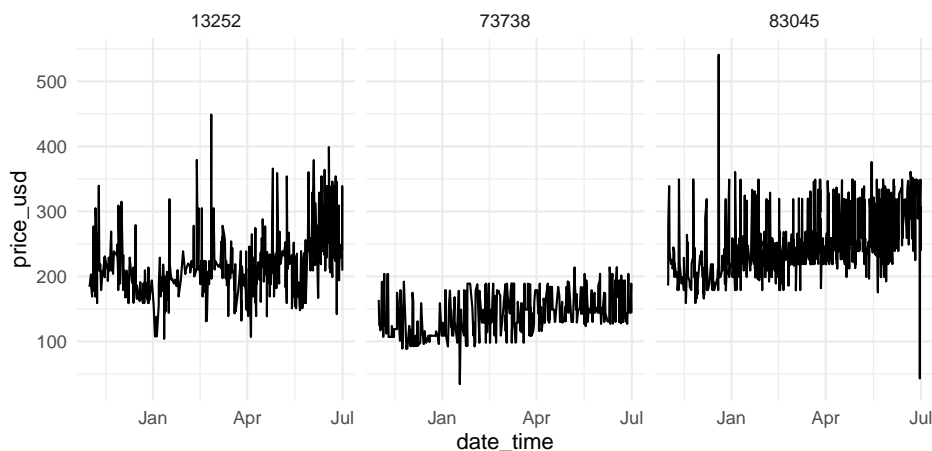


РИСУНОК 1.5. Динамика суточной стоимости номеров в трех гостиницах.
Источник: [Expedia](#)