

УДК 330.4:519.237

ИСПОЛЬЗОВАНИЕ ПАКЕТА PROPHET В ПРОГНОЗИРОВАНИИ ВРЕМЕННЫХ РЯДОВ

Орлова И.В.*ФГБОУ ВО «Финансовый университет при Правительстве РФ», Москва, e-mail: IVOrlova@fa.ru*

В статье рассматривается задача прогнозирования временных рядов с различной частотой данных с использованием пакета Prophet. Он доступен как в Python, так и в R. В Prophet удачно реализована возможность получать достаточно точные прогнозы как полностью автоматизированным методом, так и с учетом экспертных оценок. Введение интуитивных параметров, которые могут быть неоднократно скорректированы, позволяет эффективно настроить модель, не зная деталей лежащей в основе математической модели, с помощью ввода экспертом легко интерпретируемых параметров. Модель временных рядов, реализованная в пакете, включает не только компоненты тренда и сезонности, но компоненту, учитывающую влияние праздников. Методика построения прогнозов рассмотрена на еженедельных данных по курсу акций Сбербанка за период с июня 2014 г. по 15 февраля 2021 г. Оценивание параметров подгоняемой модели выполняется с использованием принципов байесовской статистики. Для задачи прогнозирования стоимости акций Сбербанка использовалась кусочно-линейная трендовая модель, в которой Prophet обнаруживает точки изменения тренда и позволяет тенденции соответствующим образом адаптироваться. С целью улучшения качества подгонки было построено четыре модели, в которых изменялись параметры функции prophet. Выбор оптимальной модели был выполнен с помощью перекрестной проверки моделей временных рядов с помощью функции cross-validation. При сравнении истинных значений стоимости акций Сбербанка и прогнозных значений, полученных с помощью оптимальной модели, были получены неплохие значения характеристик точности. В работе даны некоторые практические рекомендации по применению пакета Prophet при прогнозировании временных рядов высокочастотных данных.

Ключевые слова: временные ряды, пакет Prophet, прогнозирование

USING THE PROPHET PACKAGE IN TIME SERIES FORECASTING

Orlova I.V.*Financial University under the Government of the Russian Federation, Moscow, e-mail: ivorlova@fa.ru*

The article deals with the problem of forecasting time series with different data frequency using the Prophet package. It is available in both Python and R. Prophet has successfully implemented the ability to obtain fairly accurate forecasts both by a fully automated method and taking into account expert estimates. The introduction of intuitive parameters that can be repeatedly adjusted allows you to effectively tune the model without knowing the details of the underlying mathematical model by entering easily interpreted parameters by an expert. The time series model implemented in the package includes not only trend and seasonality components, but also a component that takes into account the influence of holidays. The forecasting methodology is considered on weekly data on the Sberbank share price for the period from June 2014 to February 15, 2021. Estimation of the parameters of the fitted model is performed using the principles of Bayesian statistics. For the task of forecasting the value of Sberbank shares, a piecewise linear trend model was used, in which Prophet detects points of trend change and allows the trend to adapt accordingly. In order to improve the quality of the fit, four models were built in which the parameters of the prophet function were changed. The choice of the optimal model with was performed by cross-validating time series models using the cross-validation function. When comparing the true values of the Sberbank share value and the predicted values obtained using the optimal model, good values of the accuracy characteristics were obtained. The paper provides some practical recommendations for using the Prophet package for forecasting time series of high-frequency data.

Keywords: time series, Prophet package, forecasting

Ухудшение экономической ситуации, связанное с распространением коронавирусной инфекции, резкие темпы сокращения деловой активности привели к необходимости анализировать состояние экономики с недельной и даже дневной частотой. Для формирования обоснованных решений в условиях дефицита времени аналитики обратили внимание на инструментарий, способный к обработке высокочастотных данных.

Prophet – это библиотека прогнозирования временных рядов с открытым исходным кодом, разработанная командой Facebook. Он доступен как в Python,

так и в R. Например, в библиотеку заранее включены данные, способные влиять на прогноз, например праздничные дни, и их использование может существенно повлиять на прогнозы.

Цель Prophet – создавать высококачественные прогнозы для принятия решений, не требуя от пользователя экспертных знаний по прогнозированию временных рядов. Пользователь может интуитивно вмешаться в процесс построения модели, введя известные параметры, такие как точки изменения тренда из-за введения нового продукта или значения насыщения тренда из-за емкости.

Prophet способен производить надежные и точные прогнозы (часто выполняемые лучше, чем другие распространенные методы прогнозирования), обрабатывая почасовые, ежедневные или еженедельные данные с минимальными усилиями, позволяя применять знания предметной области с помощью легко интерпретируемых параметров.

Модель прогнозирования временных рядов, реализованная в Prophet, дает возможность учитывать плохо формализуемые свойства прогнозируемого временного ряда, позволяя применять знания предметной области с помощью ввода экспертом легко интерпретируемых параметров.

Введение интуитивных параметров, которые могут быть неоднократно скорректированы, позволяет эффективно настроить модель, не зная деталей лежащей в основе математической модели.

Цель исследования: прогнозирование стоимости акций Сбербанка на основании еженедельных данных с помощью пакета Prophet.

Материалы и методы исследования

В исследовании использованы еженедельные данные по курсу акций Сбербанка за период с января 2014 г. по 15 февраля 2021 г. [1].

Из исследования динамики котировок за несколько лет вытекает, что хотя курс акций Сбербанка постоянно меняется, но в целом превалирует тенденция к росту стоимости акций. Поэтому его ценными бумагами интересуются инвесторы, которые хотят заработать на изменении курса. С этой целью они должны регулярно отслеживать котировки акций. Наибольшее значение в наблюдаемом периоде было достигнуто в декабре 2020 г., когда за одну ценную бумагу давали 283,7 руб. Затем, после небольшого снижения, тенденция вновь стала положительной [2].

Общий вид модели. В Prophet при построении аддитивной регрессионной модели временного ряда каждый уровень временного ряда можно представить как функцию трех компонент модели: тренд, сезонность и праздники.

$$y(t) = g(t) + s(t) + h(t) + \varepsilon(t), \quad (1)$$

$g(t)$ – кусочно-линейный или логистический тренд кривой роста, который отражает долговременную тенденцию развития, $s(t)$ – периодические колебания (например, недельная и годовая сезонность), $h(t)$ – влияние праздников, $\varepsilon(t)$ представляет любые изменения, которые не учитываются моделью; предполагается, что $\varepsilon(t)$ нормально распределено.

В спецификации этой аддитивной модели в качестве регрессора используется время, но возможны комбинации из нескольких линейных и нелинейных функций времени. В такой постановке легко добавляются новые компоненты, например, при выявлении нового источника сезонности. Преимуществом модели является возможность пользователя интерактивно изменить параметры модели.

Оценивание параметров модели выполняется с использованием принципов байесовской статистики. Для этого применяется платформа вероятностного программирования Stan [3].

Stan – это современная платформа для статистического моделирования и высокопроизводительных статистических вычислений. Многие исследователи используют Stan для статистического моделирования, анализа данных и прогнозирования в социальных, биологических и физических науках, инженерии и бизнесе. Stan реализует основанные на градиенте алгоритма Монте-Карло цепи Маркова (MCMC) для байесовского вывода, стохастические, основанные на градиенте вариационные байесовские методы для приближенного байесовского вывода и оптимизацию на основе градиента для оценки максимального правдоподобия со штрафными санкциями. Stan реализует эффективную оценку вероятностных моделей на больших наборах данных. Математическая библиотека Stan предоставляет дифференцируемые вероятностные функции и линейную алгебру.

Ядро процедуры Prophet реализовано на вероятностном языке программирования Stan. При установке Prophet Stan устанавливается вместе с библиотеками R или Python. Дополнительные пакеты R обеспечивают линейное моделирование на основе выражений, апостериорную визуализацию и перекрестную проверку без исключения.

Модель тренда. Тренд может быть либо линейным, либо насыщаемым. В обоих случаях скорость роста тренда не постоянна и может меняться несколько раз, в точках, вычисляемых автоматически. При моделировании тренда используются две модели: кусочно-линейная регрессионная модель и модель логистического роста

$$g(t) = \frac{C}{1 + \exp(-k(t - m))}, \quad (2)$$

где C – предельное значение, верхний порог, максимальная емкость для $g(t)$; k – скорость роста и m – параметр смещения, позволяющий «сдвигать» функцию вдоль оси времени [4].

Нелинейный, насыщенный рост. Есть два важных аспекта модели роста, которые не учтены в (2). Первый – предельное значение C не является постоянной величиной и в Prophet заменено на изменяемую по времени емкость $C(t)$. Второй – темпы роста не являются постоянными, поэтому модель должна быть в состоянии включать различные темпы, с тем чтобы соответствовать историческим данным. Трендовые изменения включаются в модель роста, явно определяя точки изменения, в которых темп роста может изменяться. Предположим, что существуют S точек изменения в моменты времени $s_j, j = 1, \dots, S$. Определим вектор корректировки скорости $\delta \in R^S$, где δ_j – изменение скорости, происходящее в момент времени s_j . Скорость в любой момент t тогда является базовой скоростью k , плюс все корректировки до этого момента $k + \sum_{j:t>s_j} \delta_j$.

Это представляется более четко, определив вектор $a(t) \in \{0,1\}^S$ таким, что

$$a_j(t) = \begin{cases} 1, & \text{if } t \geq s_j \\ 0, & \text{в противном случае.} \end{cases}$$

Тогда скорость в момент времени t равна $k + a(t)^T \delta$. Когда скорость k корректируется, параметр смещения m также должен быть скорректирован для соединения конечных точек сегментов. Правильная регулировка в точке изменения j легко вычисляется как

$$\gamma_j = \left(s_j - m - \sum_{l<j} \gamma_l \right) \left(1 - \frac{k + \sum_{l<j} \delta_l}{k + \sum_{l \leq j} \delta_l} \right).$$

Тогда кусочно-логистическая модель роста

$$g(t) = \frac{C(t)}{1 + \exp\left(-\left(k + a(t)^T \delta\right)\left(t - \left(m + a(t)^T \gamma\right)\right)\right)}.$$

Важным параметром в модели является $C(t)$, или ожидаемая мощность системы в любой момент времени. Этот параметр аналитики могут изменять во время решения задачи, имея представление о размерах рынка.

Линейный тренд с точками изменения. Для задач прогнозирования, которые не демонстрируют насыщающего роста, используется кусочно-линейная трендовая модель

$$g(t) = \left(k + a(t)^T \delta\right)t + \left(m + a(t)^T \gamma\right),$$

где k – скорость роста, δ – корректировка скорости, m – параметр смещения, а γ_j устанавливается так, чтобы сделать функцию непрерывной.

В реальных временных рядах часто происходят резкие изменения траекторий. По умолчанию Prophet автоматически обнаруживает эти точки изменения и позволяет тенденции соответствующим образом адаптироваться. Точки изменения могут быть указаны аналитиком с использованием известных дат событий, изменяющих рост, или могут быть автоматически выбраны системой.

Сезонность. Временные ряды экономических показателей часто имеют многопериодную сезонность. Например, 5-дневная рабочая неделя может оказывать влияние на временные ряды, которые повторяются каждую неделю, в то время как график отпусков и школьные каникулы могут оказывать влияние, которое повторяется каждый год. Чтобы моделировать и прогнозировать эти эффекты, необходимо добавить в модель сезонную компоненту.

Если данные содержат наблюдения более чем за два года, Prophet автоматически добавляет в модель компоненты годовой и недельной сезонности. Помимо этого, пользователи имеют возможность добавить и другие сезонные компоненты, например месячную, квартальную. Годовая сезонность моделируется с помощью ряда Фурье.

Праздники и события. Праздники и события приводят к несколько предсказуемым потрясениям для многих временных рядов и часто не укладываются в рамки периодической модели, поэтому их последствия не очень хорошо моделируются плавным циклом. Например, День Благодарения в Соединенных Штатах происходит в четвертый четверг ноября. Суперкубок, одно из крупнейших телевизионных событий в США, происходит в воскресенье в январе или феврале, что трудно объявить программным. Многие страны по всему миру имеют праздники, которые следуют лунному календарю. Влияние конкретного праздника похоже из года в год, поэтому важно включить его в прогноз. В пакете есть список прошлых и будущих событий, идентифицированных событием или уникальным названием праздника. Для данной проблемы прогнозирования используется объединение глобального набора праздников и страновой специфики. Включение этого списка праздников в модель предполагает, что последствия праздников являются независимыми.

Рассмотрим возможности библиотеки Prophet для прогнозирования и посмотрим на качество предсказания.

Результаты исследования и их обсуждение

Prophet лучше всего работает с данными ежедневной периодичности по край-

ней мере за один год исторических данных. Можно использовать Prophet для прогнозирования с использованием данных за день или месяц, для нашей задачи мы будем использовать еженедельную периодичность.

Если в исходных данных встречаются наблюдения, которые не относятся к сезонным колебаниям, но проявляются в виде резких скачков или спадов, то, скорее всего, – это выбросы, аномальные наблюдения [5]. При работе с Prophet выбросы можно удалить, пропущенные значения будут автоматически обработаны.

Несмотря на то, что Prophet относительно устойчив к отсутствующим данным, важно убедиться, что во временных рядах не пропущено значительное количество наблюдений. Если во временных рядах отсутствует большое количество наблюдений, следует рассмотреть возможность использования метода повторной выборки или прогнозирования данных с меньшей частотой (например, составление ежемесячных прогнозов с использованием ежемесячных наблюдений). Часто при прогнозировании выбирают конкретный тип преобразования для удаления шума перед вводом данных в модель прогнозирования (например, логарифмирование значений).

Входные данные для Prophet должны быть подготовлены в виде фрейма данных с двумя столбцами: ds, а также Y. Столбец (метка даты) ds должен иметь формат ГГГГ-ММ-ДД для даты или ГГГГ-ММ-ДД ЧЧ:ММ:СС для метки времени. Столбец Y должен быть числовым и содержать наблюдения временного ряда, который мы прогнозируем. На рис. 1 приведен график динамики стоимости акций Сбербанка с января 2014 г. по 15 февраля 2021 г.

Так выглядят данные (начало и конец временного ряда акций Сбербанка, подго-

товленные для построения модели прогнозирования в Prophet:

```
> head(SB)
      ds      Y
1 2014-01-06  99.20
2 2014-01-13 101.17
3 2014-01-20  99.70

> tail(SB)
      ds      Y
370 2021-02-01 271.70
371 2021-02-08 266.00
372 2021-02-15 270.41
```

Разделим нашу выборку на две подвыборки: обучающую (SB_train) и контролирующую (SB_test). По обучающей выборке будем строить модели с разными параметрами, а контролирующую выборку будем использовать для проверки адекватности лучшей модели. Для снижения дисперсии данные были прологарифмированы.

Для построения моделей будем использовать функцию prophet (). Основные аргументы функции prophet ():

df – фрейм данных, должны быть столбцы ds (тип даты) и Y, временной ряд;

growth – тип тренда, может принимать одно из двух значений: linear (линейный, присвоено по умолчанию) или logistic (логистический);

changepoints – вектор дат, в которые следует включить потенциальные точки изменения тренда (если точки не указаны, то потенциальные точки изменения выбираются автоматически);

n.changepoints – предполагаемое количество точек излома тренда (25 по умолчанию);

changepoint.range – доля исторических данных (начиная с первого наблюдения), по которым будут оценены точки излома (по умолчанию она составляет 80% наблюдений);

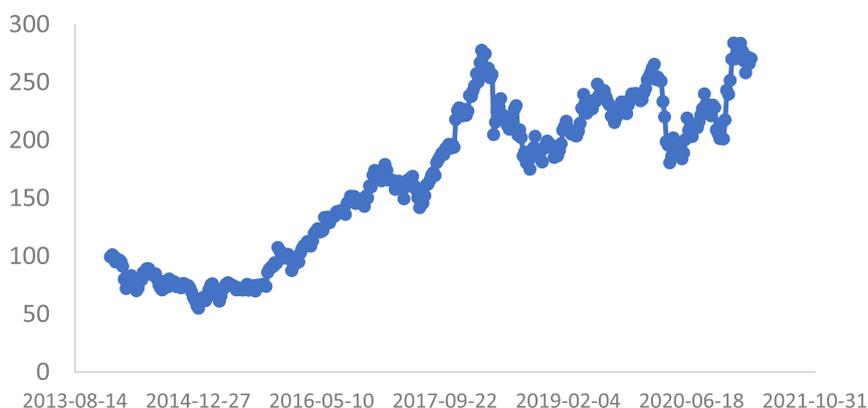


Рис. 1. График динамики стоимости акций Сбербанка

*. **seasonality** – параметры настройки различных типов сезонности (годовой, месячной, недельной, дневной);

holidays – (необязательно) фрейм данных со столбцами holiday (символ) и ds (тип даты) для включения в модель праздников и других влияющих дат.

interval.width – число, определяющее ширину доверительного интервала для предсказанных моделью значений (0,8 по умолчанию, что соответствует 80%-ному интервалу).

С другими параметрами, как всегда, можно ознакомиться в документации для этой функции.

Первую модель **mod1** построим, указав только фрейм данных **SB_train**, остальные параметры по умолчанию:

```
> mod1 <- prophet (SB_train).
```

Для получения прогноза по построенной модели с помощью функции **predict()** сначала надо подготовить таблицу с датами, охватывающими горизонт прогнозирования, используя для этого функцию **make_future_dataframe()**.

```
> future_SB <- make_future_dataframe (mod1, periods = 20, freq='weeks')
> forecast_mod1 <- predict (mod1, future_SB).
```

В объекте **forecast_mod1** хранятся фактические значения y_i , предсказанные значения отклика \hat{y}_i , (**yhat**) а также верхние (**yhat_upper**) и нижние (**yhat_lower**) 80%-ные границы доверительных интервалов соответствующих величин (прологарифмированные).

```
> forecast_mod1 %>%
```

```
+ dplyr::select(yhat, yhat_lower, yhat_upper) %>% head()
```

	yhat	yhat_lower	yhat_upper
1	4.549387	4.462838	4.626928
2	4.551238	4.468548	4.630405
3	4.544821	4.457967	4.626385

```
> forecast_mod1 %>%
```

```
+ dplyr::select(yhat, yhat_lower, yhat_upper) %>% tail()
```

	yhat	yhat_lower	yhat_upper
370	5.398769	5.297796	5.503412
371	5.409993	5.311765	5.506838
372	5.415317	5.312299	5.530737

На рис. 2 приведен график результатов моделирования и прогнозирования стоимости акций Сбербанка по модели **mod1**, полученный с помощью функции **plot(mod1, forecast_mod1)**. Точки на рис. 2 соответствуют (логарифмированным) значениям стоимости акций Сбербанка, голубая сплошная линия – это предсказанные моделью значения стоимости, а огибающая эту линию «лента» обозначает 80%-ные доверительные границы предсказанных значений. Прогнозные значения стоимости акций на период прогнозирования видны в правой части графика.

На графике, полученном с помощью функции **plot(mod1, forecast_mod1) + add_changepoints_to_plot(mod1)** (рис. 3) по сравнению с рис. 2 добавлены точки излома тренда, обозначенные штриховыми красными линиями и тренд – сплошная красная линия. В этой модели точки излома обнаружены автоматически.

Проверка качества модели на обучающей выборке (рис. 4) показала, что большинство истинных значений (красные точки) не попали в доверительный интервал (голубая полоса) предсказанных значений.

В дальнейшем было построено четыре модели, в которых изменялись параметры функции **prophet()** с целью улучшения качества подгонки. Среди настраиваемых параметров в **Prophet** наиболее важным является коэффициент масштаба распределения весов для контрольных точек (**changepoint_prior scale**).

changepoint_prior_scale – определяет гибкость тренда и, в частности, насколько сильно тренд меняется в точках изменения тренда. Значение по умолчанию 0,05 используется для многих временных рядов, но его можно изменить. Коэффициент масштаба распределения весов для контрольных точек отражает объём «уделенного внимания» точкам изменения курса

акций. Это используется для контроля недообучения и переобучения модели.

`seasonality_prior_scale` – параметр контролирует гибкость сезонности.

Большое значение позволяет сезонности соответствовать большим колебаниям, маленькое значение уменьшает величину сезонности; диапазон $[0,01, 10]$.

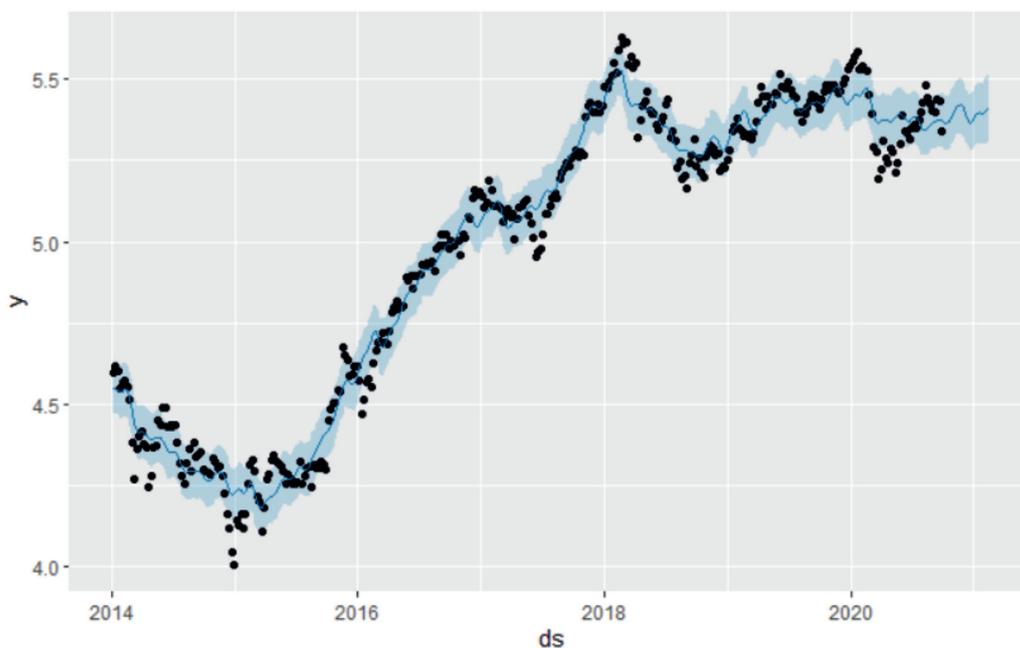


Рис. 2. Результаты моделирования и прогнозирования стоимости акций Сбербанка по модели `mod1`



Рис. 3. Тренд и точки излома тренда по модели `mod1`

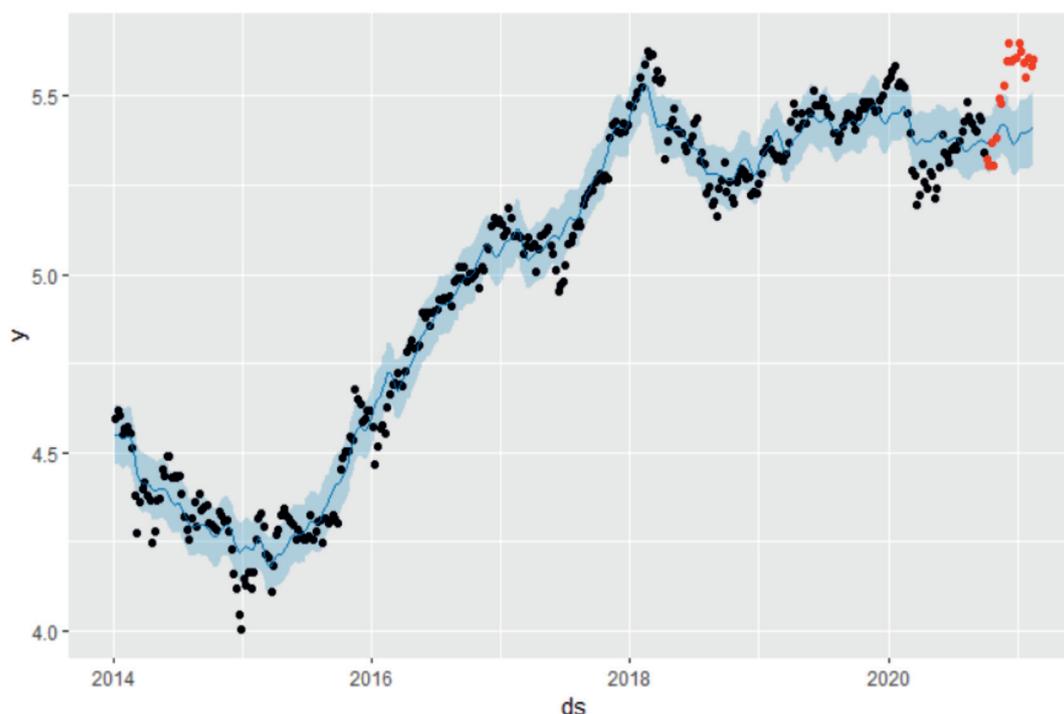


Рис. 4. Сравнение истинных значений стоимости акций Сбербанка и прогнозных значений, полученных с помощью модели `mod1`

После построения нескольких моделей для прогнозирования стоимости акций Сбербанка надо выбрать лучшую из них. В пакете Prophet для выбора оптимальной модели с помощью перекрестной проверки моделей временных рядов реализован метод «имитированных исторических прогнозов» (Simulated Historical Forecasts, SHF) [6], который выполняется с помощью функции `cross_validation()`.

Аргументы функции:

- `model` – модельный объект;
- `horizon` – длина прогнозного горизонта в каждом блоке данных, используемом для выполнения перекрестной проверки;
- `units` – название единицы измерения времени (например, «days», «hours»);

- `initial` – длина начального отрезка с обучающими данными в первом блоке.

Результатом `cross_validation` является фрейм данных с истинными значениями y_i и значениями прогноза \hat{y}_i вне выборки для каждой моделируемой даты прогноза и для каждой даты отсечения. В частности, прогноз делается для каждой наблюдаемой точки между `cutoff` и `cutoff + horizon`. Эти данные могут быть использованы для вычисления характеристик качества подгонки моделей.

Далее мы выполняем для оценки качества прогнозирования перекрестную проверку на горизонте 20 недель, начиная с 180 недель обучающих данных в первом пороговом значении, а затем делая прогнозы каждые 20 недель для каждой из построенных моделей `mod1 – mod4`.

```
M1_cv <- cross_validation(mod1, initial = 180,
                           period = 20,
                           horizon = 20,
                           units = "weeks")
```

На основании данных, полученных в результате работы функции `cross_validation()`, вычисляются показатели, характеризующие качество предсказаний моделей (таблица).

В приведенных формулах n – количество наблюдений, y_i – это фактическое, а \hat{y}_i – предсказанное значения прогнозируемой переменной.

Показатели точности прогноза

Наименование	Показатель	Формула	Описание
Mean error	MSE	$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$	Среднее отклонение
Root mean squared error	RMSE	$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$	Среднеквадратичное отклонение
Mean absolute error	MAE	$MAE = \frac{1}{n} \sum_{i=1}^n y_i - \hat{y}_i $	Среднее абсолютное отклонение
Mean absolute percentage error	MAPE	$MAPE = \frac{1}{n} \sum_{i=1}^n \left \frac{y_i - \hat{y}_i}{y_i} \right 100\%$	Среднее абсолютное процентное отклонение
Coverage	Coverage	Процент реальных значений смоделированных переменных, которые находятся в пределах доверительной вероятности прогноза.	Покрытие

Для оценки качества моделей использовалась функция `performance_metrics()`.

Аргументы функции `performance_metrics()`:

- `df` – таблица, полученная с помощью функции `cross_validation()`;
- `metrics` – вектор имён показателей качества; если этот аргумент отсутствует, то вычисляются все показатели, перечисленные в таблице;
- `rolling_window` – размер «скользящего окна», в пределах которого происходит усреднение каждой метрики (по умолчанию принимает значение 0,1). Если `rolling_window = 0`, то запрашиваемые метрики качества рассчитываются для каждой даты прогнозного горизонта, если `rolling_window = 1`, то запрашиваемые метрики качества усредняются по всему прогнозному горизонту.

Для оценки качества четырех построенных моделей вычислим средние значения всех доступных в функции `performance_metrics()` метрик для каждой из моделей-кандидатов:

```
> performance_metrics(mod1_cv,
+                       rolling_window = 1)
  horizon      mse      rmse      mae      mape  coverage
1 140 days  0.0464  0.2153  0.1711  0.0319    0.33
> performance_metrics(mod2_cv,
+                       rolling_window = 1)
  horizon      mse      rmse      mae      mape  coverage
1 140 days  0.0318  0.1758  0.1462  0.0273    0.55
> performance_metrics(mod3_cv,
+                       rolling_window = 1)
  horizon      mse      rmse      mae      mape  coverage
1 140 days  0.0405  0.2013  0.1757  0.0327    0.22
>
> performance_metrics(mod4_cv,
+                       rolling_window = 1)
  horizon      mse      rmse      mae      mape  coverage
1 140 days  0.0388  0.1971  0.1463  0.0273    0.64
```

Модель `mod4` оказалась лучше других по показателю `coverage`, 64% фактических значений моделируемой переменной находятся в пределах доверительной вероятности прогноза. По другим показателям модель `mod4` ненамного уступает модели `mod2`.

На рис. 5 приведены результаты моделирования и прогнозирования по выбранной нами в качестве оптимальной модели `mod4`. Эта модель дала неплохой результат: 64% истинных значений стоимости акций Сбербанка оказались в пределах 80%-ной доверительной полосы.

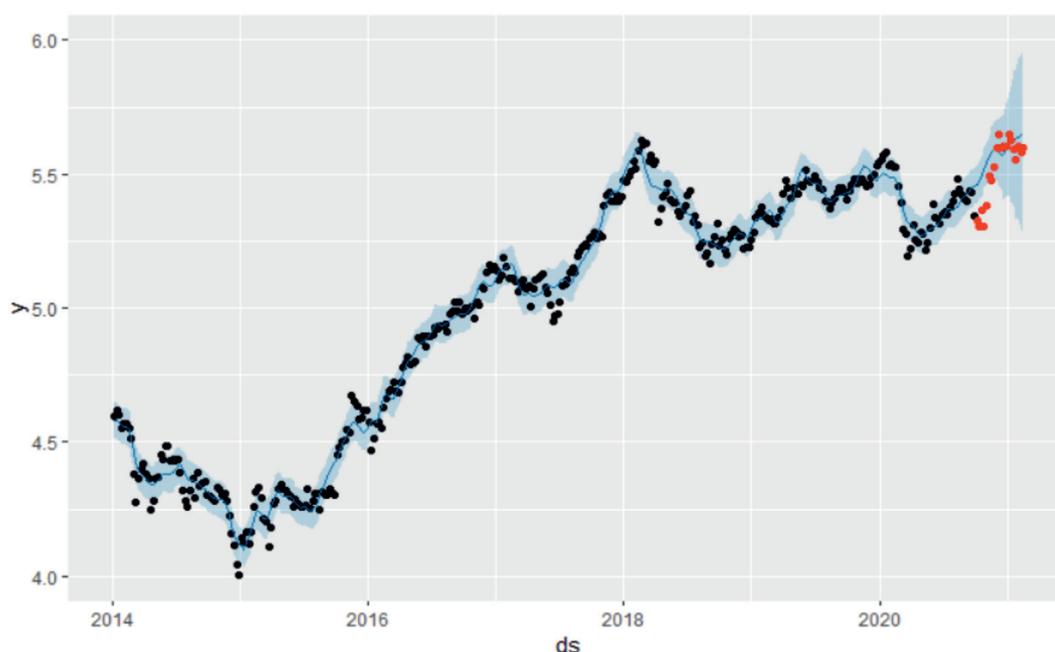


Рис. 5. Сравнение истинных значений стоимости акций Сбербанка и прогнозных значений, полученных с помощью оптимальной модели mod4

Заключение

В результате проведенного исследования можно сделать вывод о целесообразности использования пакета Prophet для прогнозирования временных рядов экономических показателей. Особенно этот программный продукт может быть полезен практикам, не имеющим глубоких знаний в области эконометрического моделирования, позволяя применять знания предметной области и получать достаточно точные прогнозы.

Список литературы

1. Финам.ру – финансовый портал [Электронный ресурс]. URL: <https://www.finam.ru/profile/moex-akcii/sberbank/export/?market=1&em=3&token=&code=SBER&apply=0&>

[f=24&mf=6&yf=2020&from=24.07.2020&dt=24&mt=6&yt=2020&to=24.07.2020&p=7&f=SBER_200724_200724&e=.txt&cn=SBER&dtf=1&tmf=1&MSOR=1&mstime=on&mstimever=1&sep=1&sep2=1&datf=1&at=1](https://www.finam.ru/profile/moex-akcii/sberbank/export/?market=1&em=3&token=&code=SBER&apply=0&f=24&mf=6&yf=2020&from=24.07.2020&dt=24&mt=6&yt=2020&to=24.07.2020&p=7&f=SBER_200724_200724&e=.txt&cn=SBER&dtf=1&tmf=1&MSOR=1&mstime=on&mstimever=1&sep=1&sep2=1&datf=1&at=1) (дата обращения: 16.02.2021).

2. Гид по банкам [Электронный ресурс]. URL: <https://bankigid.net/dinamika-rosta-akcij-sberbanka-za-poslednie-5-let/#i-2> (дата обращения: 16.02.2021).

3. Stan Development Team (2020). RStan: the R interface to Stan. R package version 2.21.2. [Electronic resource]. URL: <http://mc-stan.org/> (date of access: 16.02.2021).

Taylor S.J., Letham B. Forecasting at scale. The American Statistician, 2017. Vol. 72(1). P. 37–45.

4. Орлова И.В. Выявление аномальных наблюдений при анализе динамики безработицы в Российской Федерации // *Фундаментальные исследования*. 2020. № 5. С. 142–148.

5. Мастицкий С.Э. Анализ временных рядов с помощью R. Электронная книга. 2020. [Электронный ресурс]. URL: <https://ranalytics.github.io/tsawithr> (дата обращения: 16.02.2021).