

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГАОУ ВО НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук, Факультет экономических наук  
Образовательная программа «Экономика и анализ данных»

УДК 519.642.4

**Отчет об исследовательском проекте на тему:**  
**Предсказание спроса в ритейле для эффективного управления запасами**  
(итоговый, этап 2)

**Выполнил студент:**

группы БЭАД231, 2 курса

Чайковский Григорий Тимофеевич

**Принял руководитель проекта:**

Богуцкий Денис Александрович

Проектно-исследовательская лаборатория «ИИ в математических финансах» ФКН

Стажер-исследователь

Москва 2025

## Содержание

<b>Введение</b>	<b>3</b>
<b>ARIMA</b>	<b>3</b>
<b>VAR</b>	<b>4</b>
<b>RNN</b>	<b>4</b>
<b>LSTM</b>	<b>4</b>
<b>Постановка задачи</b>	<b>6</b>
<b>Неформальная постановка задачи</b>	<b>6</b>
<b>Формальная постановка задачи</b>	<b>6</b>
<b>Экономические предпосылки</b>	<b>7</b>
<b>Предлагаемая модель</b>	<b>8</b>
<b>Описание модели</b>	<b>8</b>
<b>Задача оптимизации</b>	<b>8</b>
<b>Применение модели</b>	<b>10</b>
<b>Используемые данные</b>	<b>10</b>
<b>Подбор гиперпараметров</b>	<b>10</b>
<b>Интерпретация результатов</b>	<b>11</b>
<b>Мудрость толпы</b>	<b>13</b>
<b>Моделирование спроса с LSTM</b>	<b>15</b>
<b>Выводы</b>	<b>16</b>
<b>Источники</b>	<b>17</b>
<b>Список литературы</b>	<b>17</b>
<b>Приложения</b>	<b>17</b>

## Введение

Спецификой ритейла является высокая волатильность спроса и сильно ограниченное время реализации товаров [1]. В этой связи прогнозирование спроса является критически важной задачей для бизнеса, для решения которой используются методы статистики, машинного обучения и нейронных сетей [2].

Прогнозирование временных рядов, в частности задача предсказания спроса, является классической задачей регрессии. Требуется найти отображение, которое данному временному ряду - последовательности  $y_1, \dots, y_t$  сопоставит набор предсказаний -  $\bar{y}_{t+1}, \dots, \bar{y}_{t+H}$ , минимизировав разницу между предсказаниями и настоящими значениями  $y_1, \dots, y_{t+H}$ .

Задачу предсказания временных рядов решают разными способами. Во-первых, это «модели временных рядов», предполагающие различные виды линейной зависимости между членами временного ряда. Во-вторых, это методы машинного обучения, такие как линейная регрессия, решающие деревья, kernel regression, градиентный бустинг. Наконец, в последние годы для предсказания временных рядов стали активно использоваться нейрости (artificial neural networks) [1].

Особенностью спроса как временного ряда является то, что фирма может влиять на некоторые из признаков, задающих спрос, а именно на признаки связанные с ценой. В этой связи методы прикладной математики активно применяются для решения задач оптимизации, связанных с максимизацией прибыли при известной функции спроса. В частности, модели спроса основанные на эластичности по цене (elasticity based demand function, EDF) [7].

Поскольку задача непосредственно связана с выгодой для бизнеса, открытых исследований на эту тему немного: компании, продающие свои консультационные услуги по выставлению оптимальных цен, не заинтересованы в открытости методов вычисления эластичностей спроса.

В этом исследовании предлагается использование моделей временных рядов SARIMAX для решения задачи предсказания спроса и оптимизации прибыли ритейлера.

## ARIMA

В качестве базовой модели выбрано Auto-regressive integrated moving average, одну из самых популярных статистических моделей для предсказания временных рядов [6]. ARIMA( $p, d, q$ ) предполагает следующую зависимость между членами временного ряда:

$$y_t = \alpha_1 y_{t-1} + \dots + \alpha_p y_{t-p} + \beta_1 \varepsilon_1 + \dots + \beta_q \varepsilon_{t-q} + \varepsilon_t \quad (1)$$

где  $y_t$  - значение в момент  $t$ ,  $\varepsilon_i$  - «белый шум» (компонента ряда, которую невозможно предсказать) в момент  $t$

## VAR

Вместе с тем, кажется разумным, что продажи одних товаров могут влиять на продажи других в разной степени. Как минимум, продажи той же категории товаров в предыдущие периоды влияют на продажи этой категории сегодня больше, чем продажи других категорий. Какие-то товары могут являться компонентами для других: спрос на машины приведет к увеличению спроса на бензин. Чтобы учитывать эти взаимосвязи, можно применить модель векторной авторегрессии (VAR), что и было сделано.

$$y_t = c + A_1 y_{t-1} + A_2 y_{t-2} + \dots + A_p y_{t-p} + \varepsilon_t, \quad (2)$$

$$A \in R^{k \times k},$$

$$y_i, c, \varepsilon_i \in R^k$$

Данная модель также широко применяется для решения поставленной задачи [2, 3].

## RNN

Архитектура RNN предназначена для работы с последовательностями и решает более широкий класс задач, в том числе прогнозирование временных рядов. Отличительной особенностью является наличие скрытого состояния - вектора, хранящего информацию о контексте, то есть всех предыдущих элементах последовательности. Однако стандартные RNN сталкиваются с проблемой затухающих градиентов (vanishing gradients), из-за которой ее способность работать с длинными последовательностями сильно ограничена.

## LSTM

Для решения проблемы затухающих градиентов была предложена архитектура долгосрочной кратковременной памяти (Long Short-Term Memory, LSTM), предложенная Хохрайтером и Шмидхубером в 1997 году. Блоки LSTM хранят свое скрытое состояние отдельно от долгосрочной памяти модели, а механизмы «запоминания» и «забывания» - вентили (gates) - контролировать, что будет храниться в скрытом состоянии.

## Постановка задачи

### Неформальная постановка задачи

Как уже было сказано, задача регрессии успешно решается различными способами (часть из которых описана выше). Вместе с тем, если рассматривать процесс деятельности ритейлера не со стороны наблюдателя, а со стороны ритейлера, возникает более общая задача - задача максимизации прибыли. Обучая модель регрессии, мы получаем ответ на вопрос: как зависит спрос от различных факторов, которые на него влияют? Среди этих факторов - цена на товар, которую на самом деле фирма устанавливает сама. Поэтому с точки зрения ритейлера можно говорить не только о поиске отображения из пространства признаков в пространство объектов, но и о поиске оптимального значения цены, которое фирме стоит выставить.

### Формальная постановка задачи

Для начала требуется решить задачу регрессии: по данной выборке  $X \in \mathbb{R}^{n \times m}$  построить отображение  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ , которое приближает функцию спроса  $y(x)$ ,  $x \in \mathbb{R}^n$ , минимизируя функцию потерь. В нашем случае в качестве loss function  $\mathcal{L}(X; \omega, b)$  выбрана MAPE - mean absolute mercentage error. Зависимость спроса от признаков предполагается линейной:

$$y = \omega X + b \quad (3)$$

И задача заключается в

$$\mathcal{L}(X; \omega, b) \rightarrow \min \omega \in \mathbb{R}^n, b \in \mathbb{R} \quad (4)$$

где

$$\mathcal{L}(X; \omega, b) = \frac{1}{m} \sum \left| \frac{\hat{x}_i - x_i}{x_i} \right| \quad (5)$$

Далее для простоты иногда будем пропускать  $b$ , потому что свободный член можно без ограничения общности воспринимать как коэффициент для признака, значения которого равно 1 у всех объектов выборки.

Введем следующие обозначения:

$p_t$  — цена товара в период  $t$ ,

$$\Delta_p = \frac{p_t - p_{t-1}}{p_{t-1}},$$

$T$  — длина сезона для сезонной модели

$a \in \mathbb{R}^n$  — все остальные признаки, на которые фирма не может влиять, (6)

$$\text{TR}_t = p_t * y_t,$$

$\text{TC}_t$  — издержки фирмы,

$$\pi = \text{TR} - \text{TC}$$

Требуется решить задачу оптимизации:

$$\pi \rightarrow \max \Delta_p \quad (7)$$

### Экономические предпосылки

Чтобы такая задача имела смысл, требуется выполнение некоторых экономических условий, среди которых есть 3 основных:

- 1) Эластичный спрос: спрос на рассматриваемый товар должен сильно реагировать на изменение цены, чтобы можно было влиять на спрос путем ее изменения.
- 2) Волатильность цен: модель предполагает изменение цен каждую неделю.
- 3) Высокая степень рыночной власти. Как известно из экономической теории, на конкурентном рынке фирма является «ценополучателем» (price-taker), поэтому не может оптимизировать свою прибыль по цене.

## Предлагаемая модель

### Описание модели

На текущий момент большинство исследователей, моделирующих спрос с помощью линейных моделей, предполагают использовать аппарат линейной регрессии. В этой работе предлагается использовать аппарат интегрированной модели авторегрессии-скользящего среднего (ARIMA) для моделирования спроса.

Базовая версия модели рассматривает спрос  $Y_t$  как временной ряд, значение которого зависит от признаков, которые можно разделить на несколько групп:

- лаговые признаки (auto-regressive features)
- ошибки модели в предыдущие периоды (moving-average features)
- сезонность
- остаточный компонент (residuals)
- $\Delta_p$  - изменение цены

В виде формулы это можно изобразить в следующем виде:

$$\begin{aligned} y_t = & \alpha_1 y_{t-1} + \dots + \alpha_k y_{t-k} \\ & + \beta_1 \varepsilon_{t-1} + \dots + \beta_{t-m} \varepsilon_{t-m} \\ & + \gamma_1 y_{t-T} + \dots + \gamma_l y_{t-lT} \\ & + \alpha \Delta_p + \beta \varepsilon_t \end{aligned} \quad (8)$$

С использованием обозначений выше:

$$y_t = \alpha \Delta_p + \beta \varepsilon_t + a^T x \quad (9)$$

Таким образом, модель можно представить в виде SARIMA( $p, d, q, (P, D, Q)_T$ ) с экзогенной переменной  $\Delta_p$ . Такую модель также называют SARIMAX - сезонная ARIMA с вектором  $x$  в качестве экзогенной переменной [8].

### Задача оптимизации

Так как оптимизация производится по  $\Delta_p$ , издержки фирмы являются константой, поэтому достаточно максимизировать величину

$$TR = p_{t-1} (1 + \Delta_p) \times y_t(\Delta_p) \quad (10)$$



Так как функция спроса предполагается линейной (см. Уравнение 9), решение задачи оптимизации выписывается явным образом:

$$\begin{aligned}
\frac{\partial \text{TR}}{\partial \Delta_p} &= p_{t-1} y_t(\Delta_p) + p_{t-1} (1 + \Delta_p) y'_t(\Delta_p) = \\
&= p_{t-1} y_t(\Delta_p) + \alpha p_{t-1} (1 + \Delta_p) = 0 \\
&\Rightarrow a^T x + \alpha \Delta_p + \alpha (1 + \Delta_p) = 0 \\
&\Rightarrow \Delta_p = \frac{-a^T x - \alpha}{2\alpha}
\end{aligned} \tag{11}$$

Отдельно требуется рассмотреть случай интегрированной модели, где  $\alpha$  является коэффициентом при  $\Delta_p - L^1 \Delta_p$ , где  $L^i$  - лаговый оператор:

$$y_t = \alpha(\Delta_p - L^1 \Delta_p) + a^T x + \beta \varepsilon_t \tag{12}$$

Задача сводится к предыдущей, так как  $-\alpha L^1 \Delta_p$  является константой.

## Применение модели

### Используемые данные

Для тестирования модели используется датасет о продажах авокадо в США с платформы [Kaggle](#). Данные о продажах авокадо подходят для нашей задачи, так как они отражают волатильные цены, которые часто меняются еженедельно в зависимости от поставок, спроса и акций ритейлеров, что позволяет моделировать влияние цен на спрос. Доступность еженедельных данных о ценах и объемах через источники вроде [USDA](#) и [Hass Avocado Board](#) делает их удобными для анализа. Хотя рынок авокадо не является чистой монополией, крупные ритейлеры или брендируемые поставщики обладают достаточной рыночной властью, чтобы экспериментировать с ценами и оптимизировать прибыль, что соответствует целям исследования.

### Подбор гиперпараметров

Для выбора гиперпараметров модели построены графики автокорреляционной и частичной автокорреляционной функций: как известно, оптимальным значением  $p$  в модели  $AR(p)$  считается последний значимый пик ACF, а оптимальным значением  $q$  в  $MA(q)$  - последний значимый пик PACF [5].

Затем процесс подбора гиперпараметров автоматизирован с помощью библиотеки `sktime`. Для подбора оптимальных значений  $p$ ,  $d$ ,  $q$  использовался `grid search` - перебор всех разумных комбинаций гиперпараметров - с кросс-валидацией типа: модель последовательно обучалась на данных за предыдущие 2 года и предсказывала на 1 неделю вперед. Для наглядности приведен пример кросс-валидации временного ряда, в котором размер тренировочной выборки на каждом шаге равен 5, а размер тестовой 3. Из-за того, что данных о недельных продажах не может быть много, такой подход особенно ценен в этой задаче, так как позволяет использовать одни и те же данные сначала в обучающей выборке, затем в тестовой.

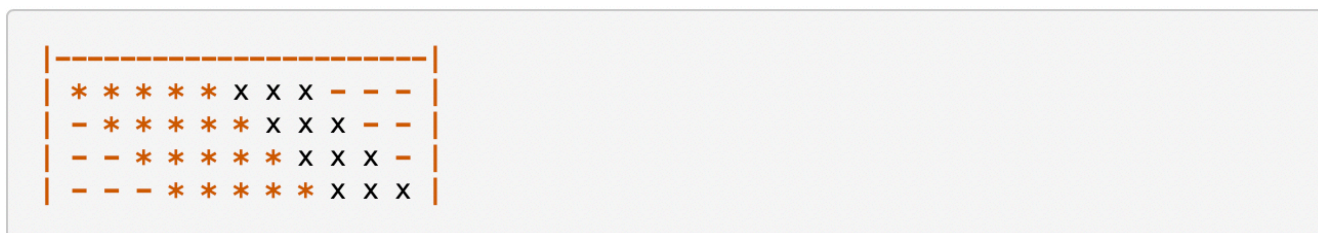


Рис. 1. \* - тренировочные данные, x - тестовые данные

	# mean_test_MeanAbsolut...	# mean_fit_time	# mean_pred_time	params	# rank_test_MeanAbsolute...
0	0.07898667486841968	0.03190028869771752	0.005260139238089323	{'order': (1, 0, 1)}	30.0
1	0.06651946943214608	0.0338038247688834	0.005202007206991829	{'order': (1, 1, 1)}	8.0
2	0.08095624307159618	0.047692988581698514	0.008135206949222705	{'order': (1, 0, 2)}	32.0
3	0.06596742603417859	0.052454820416610815	0.006035698256615935	{'order': (1, 1, 2)}	7.0
4	0.07562289866749888	0.03776886211387042	0.005508301747513229	{'order': (1, 0, 3)}	25.0
5	0.06828279659376635	0.04089504451458824	0.006132326169132158	{'order': (1, 1, 3)}	12.0
6	0.0776357142389429	0.0562961437129255	0.008942199870944023	{'order': (1, 0, 4)}	26.0
7	0.06892175193007492	0.0702294525369231	0.009511814581166053	{'order': (1, 1, 4)}	15.0
8	0.08709215919542329	0.04707898130511929	0.008074593377010575	{'order': (1, 0, 5)}	35.0
9	0.06887168539524594	0.08254627441858937	0.007573035931022003	{'order': (1, 1, 5)}	14.0
10	0.07950138445787286	0.03222192377494327	0.009458334699016193	{'order': (2, 0, 1)}	31.0
11	0.06340003744221118	0.03407930154032234	0.00864600847828491	{'order': (2, 1, 1)}	3.0
12	0.12552941942521195	0.05485815376621382	0.007729886388727303	{'order': (2, 0, 2)}	42.0
13	0.0665473557009588	0.03680477302585696	0.008120169670417392	{'order': (2, 1, 2)}	9.0
14	0.07332027200591287	0.06274051580513859	0.008955590699895703	{'order': (2, 0, 3)}	20.0
15	0.06764869152872853	0.04776696984431353	0.00723103727695757	{'order': (2, 1, 3)}	11.0
16	0.0785661951091044	0.0690335962471777	0.008676508618197564	{'order': (2, 0, 4)}	29.0
17	0.0665639453131911	0.0659740963822295	0.008658030118536332	{'order': (2, 1, 4)}	10.0
18	0.08833808765483404	0.11552481319562628	0.012520767346923721	{'order': (2, 0, 5)}	36.0
19	0.0699780686110009	0.16663082907425947	0.010343653727965108	{'order': (2, 1, 5)}	18.0
20	0.07450098400978394	0.0329651493450691	0.0071334653244964005	{'order': (3, 0, 1)}	24.0
21	0.06562687162526115	0.04860687342568718	0.008227727251426286	{'order': (3, 1, 1)}	6.0
22	0.20672431560437834	0.10602658046088342	0.01056113357430902	{'order': (3, 0, 2)}	46.0
23	0.15135609961242927	0.08780074996295674	0.01116932181898376	{'order': (3, 1, 2)}	44.0
24	0.07209828761504947	0.11739158484249793	0.011191079087940783	{'order': (3, 0, 3)}	19.0
25	0.06965973552220352	0.12431492950317674	0.01504400862252404	{'order': (3, 1, 3)}	17.0
26	0.1578074895753562	0.16802441380147276	0.012036420982973329	{'order': (3, 0, 4)}	45.0
27	0.06287444206575946	0.1406654123987617	0.010238994057450828	{'order': (3, 1, 4)}	1.0
28	0.09491748366061996	0.1413278204292573	0.01032872551024474	{'order': (3, 0, 5)}	38.0
29	0.0684936683092011	0.2063632600876535	0.014018532916389662	{'order': (3, 1, 5)}	13.0
30	0.07809505718223701	0.052870991414990916	0.011499244238025156	{'order': (4, 0, 1)}	28.0
31	0.0632535312163825	0.04846820249703938	0.010283538814762542	{'order': (4, 1, 1)}	2.0
32	0.24695986903945233	0.12982458777019176	0.01372994400625085	{'order': (4, 0, 2)}	48.0
33	1.2780702491939169	0.11940995393449376	0.008097581918640384	{'order': (4, 1, 2)}	49.0
34	0.09081790538109746	0.11934567234832151	0.009547281657056561	{'order': (4, 0, 3)}	37.0
35	0.07437668139918859	0.1528365674492871	0.009695945564528992	{'order': (4, 1, 3)}	23.0
36	0.12221240121000983	0.16720033040398669	0.012380405248881414	{'order': (4, 0, 4)}	41.0
37	0.069136190177542	0.14989462215453386	0.011746349182493728	{'order': (4, 1, 4)}	16.0
38	0.09714423920175543	0.18311026303804126	0.011373606285658377	{'order': (4, 0, 5)}	39.0
39	0.08637468668931685	0.24025298141199966	0.011417211161862159	{'order': (4, 1, 5)}	34.0
40	0.07356642275230663	0.07281498559590044	0.013279343345041933	{'order': (5, 0, 1)}	22.0
41	0.06381346977046694	0.056875310468904934	0.009012563298616943	{'order': (5, 1, 1)}	5.0
42	0.11796383245268781	0.13205338365815836	0.008726528717269158	{'order': (5, 0, 2)}	40.0
43	1.6499484588702324	0.13446529741109958	0.008196668121321448	{'order': (5, 1, 2)}	50.0
44	0.0734303067353529	0.15201122845234027	0.008640353375210845	{'order': (5, 0, 3)}	21.0
45	0.06345525699072627	0.17342802726438847	0.009176330560625627	{'order': (5, 1, 3)}	4.0
46	0.20873335712713617	0.1852483162014135	0.007920030311777675	{'order': (5, 0, 4)}	47.0
47	0.07788848776510926	0.18916462064752804	0.007321708402114695	{'order': (5, 1, 4)}	27.0
48	0.13949307418288132	0.18035846105616155	0.00613684059473975	{'order': (5, 0, 5)}	43.0
49	0.0855396442515362	0.18052772418114132	0.006415965565448177	{'order': (5, 1, 5)}	33.0

Рис. 2. Результаты кросс-валидации  
Интерпретация результатов

По итогам проведенных экспериментов наилучшие предсказания получились у модели ARIMAX(3, 1, 4).

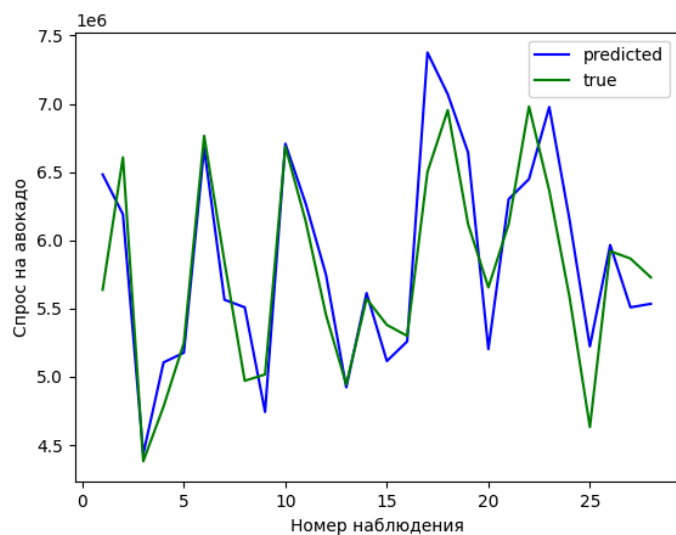


Рис. 3. результаты  $ARIMAX(p=3, d=1, q=4, X=\Delta_p)$

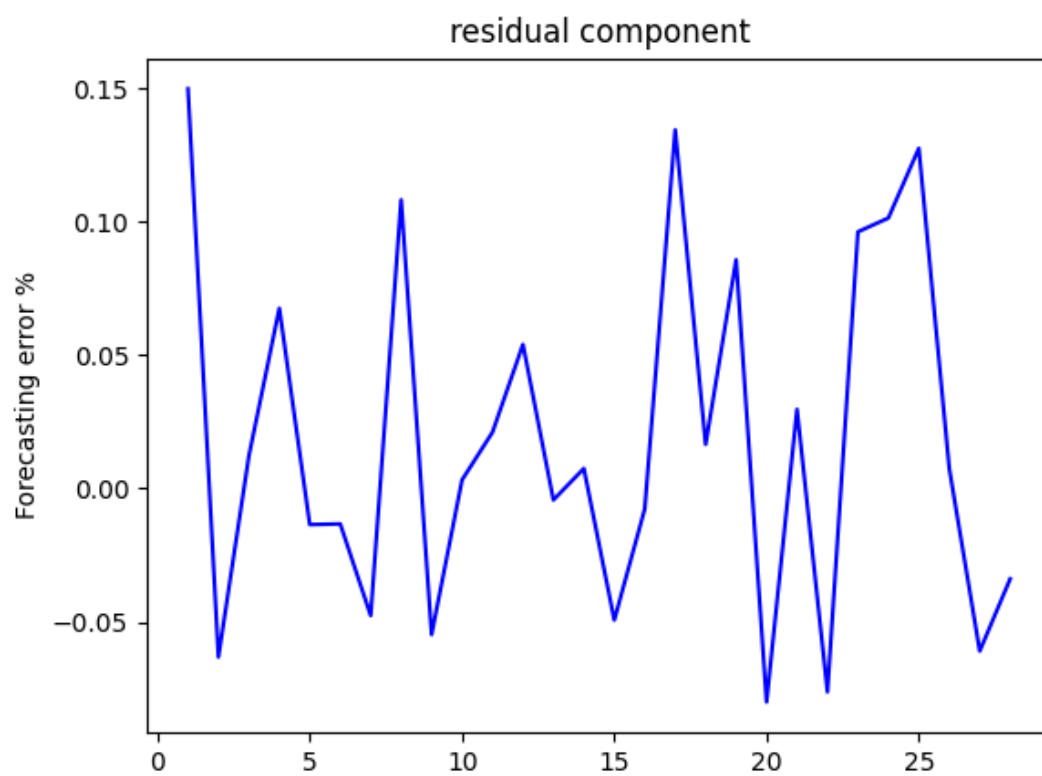


Рис. 4. Ошибки модели на тестовой выборке

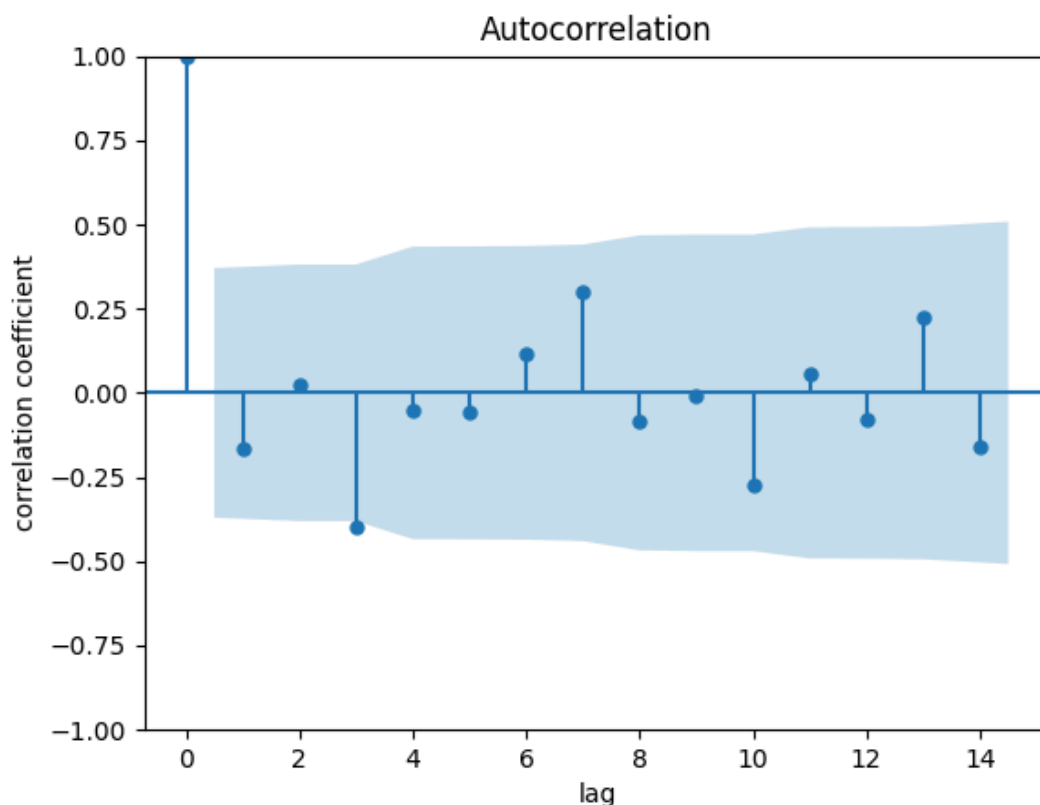


Рис. 5. Автокорреляционная функция ошибок модели

Ключевое «правильное» свойство ошибок - некоррелированность - выполняется: синей областью на графике автокорреляционной функции ряда ошибок модели отмечен интервал, в который попадают незначимые корреляции (автокоррелированность ошибок говорит о том, что модель не учитывает часть взаимосвязей между элементами временного ряда).

### Мудрость толпы

Оценка «советов» модели об установке оптимальной цены - более сложная задача - чтобы оценить ее качество, нужно было бы провести  $A/B$  тест, а такой возможности у исследователей нет.

Поэтому будем считать, что продавцы авокадо выставляют правильные или почти правильные цены на свой продукт согласно принципу «мудрости толпы» [9]: среднее большого количества независимых предсказаний должно совпадать с истинным значением. В нашем случае среднее предсказаний оптимальной цены фирмами-участниками рынка должно быть похоже на действительно оптимальное значение.

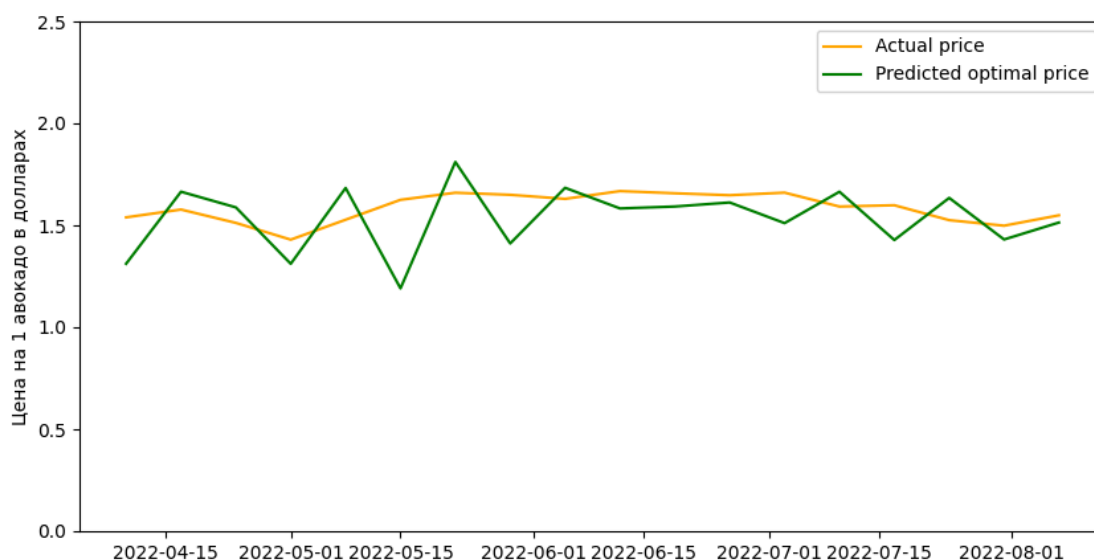


Рис. 6. Средняя цена на рынке и оптимальная цена согласно предсказанию модели

В целом модель предлагает цены, похожие на настоящие - это хорошее свойство. С другой стороны, модель предлагает более активную ценовую политику, чем в среднем по рынку - это закономерно, так как не учитывается конкуренция на рынке.

	std
Actual price	0.0628
Optimal price	0.1555

Таблица 1. Стандартное отклонение реальных и оптимальных цен  
Ниже представлен график, показывающий, какое улучшение прибыли «обещает» модель исходя из своих представлений об эластичности спроса:

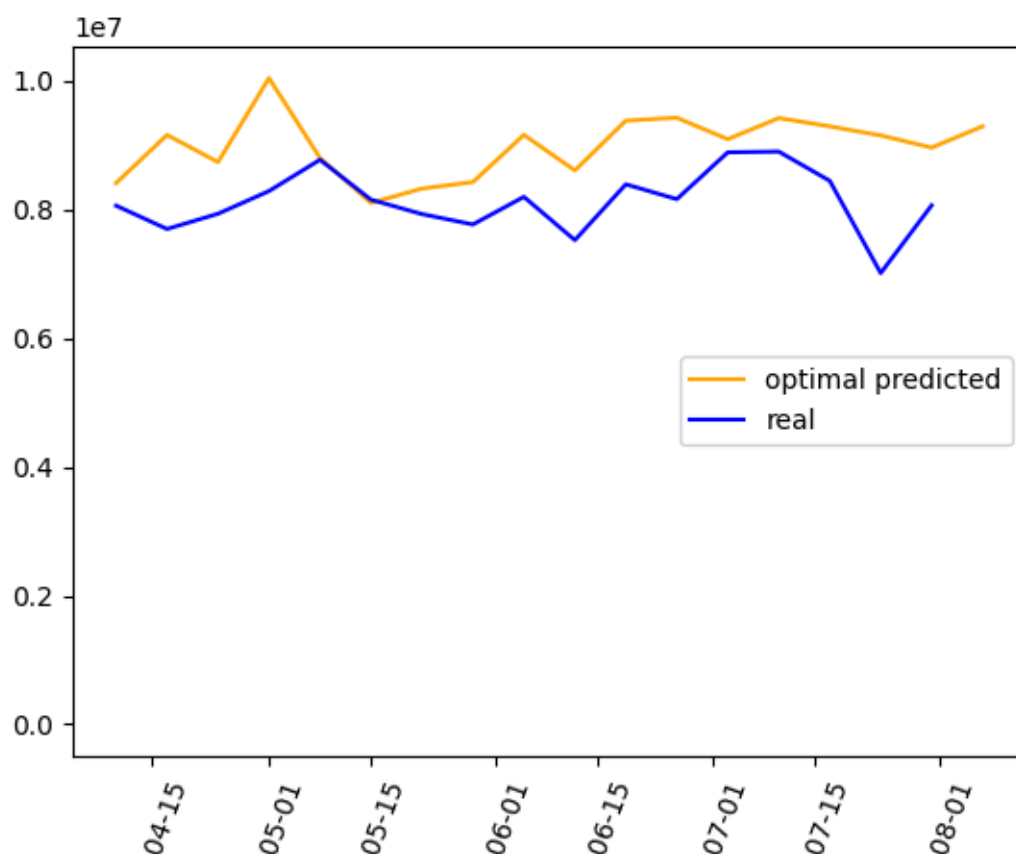


Рис. 7. Предсказанная выручка при оптимальной цене и реальной цене

На графике выше по вертикальной оси - выручка при оптимальной цене и предсказанном для этой цены спросе, по горизонтальной - соответственно, недели.

### Моделирование спроса с LSTM

Далее в качестве альтернативного подхода для сравнения используется простейшая нейросеть: 2 LSTM слоя с 200 и 30 нейронами соответственно и Dropout регуляризация, деактивирующая 20% нейронов на каждом слое (коэффициент выброса 0.2). Модель была обучена в течение 50 эпох.

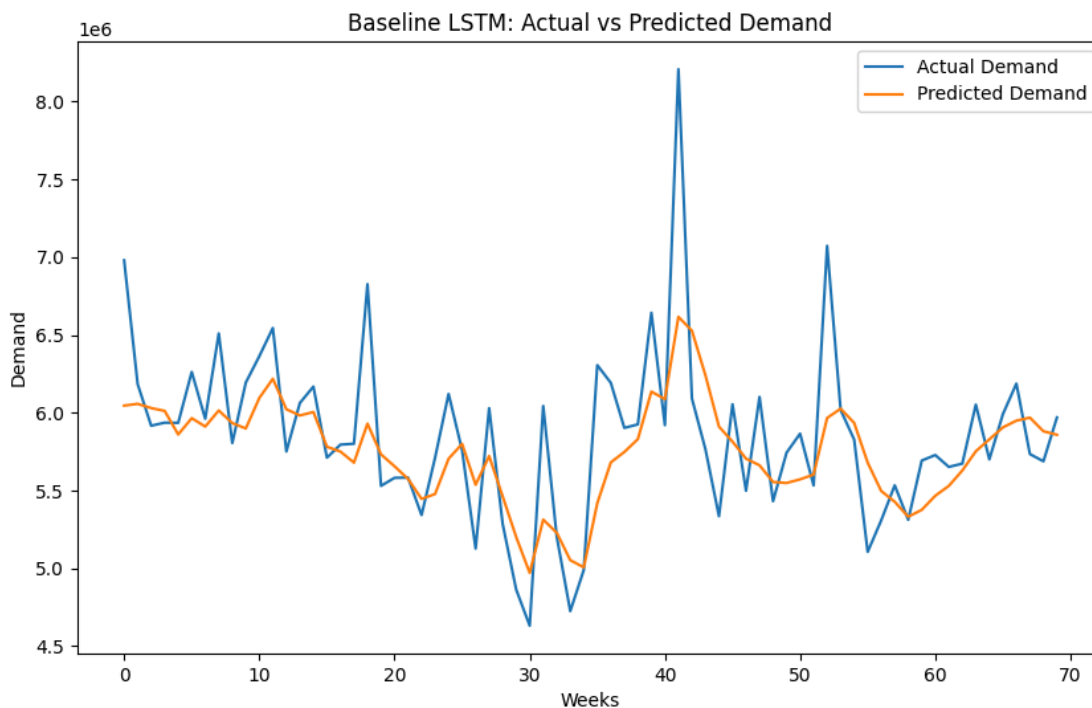


Рис. 8. Средняя цена на рынке и оптимальная цена согласно предсказанию модели

Model	test MAPE score
Naive	0.1183
SARIMAX	0.0628
LSTM	0.0381

Таблица 2. Сравнение результатов двух моделей

Как видим, даже очень простая по современным меркам нейросеть смогла превзойти классическую модель. Причем, безусловно, этот результат тоже можно улучшить. Обе модели превосходят бенчмарк-наивный предсказатель (всегда предсказывающий  $y_t = y_{t-1}$ )

## Выводы

Полученная модель дает качественные предсказания, хорошо решает задачу предсказания спроса. Решение оптимизационной задачи похоже на настоящие цены и хорошо бы подошло для фирмы-монополиста, которая может проводить активную ценовую политику.



## **Источники**

### **Список литературы**

- [1] Da Veiga CP, Da Veiga CR, Catapan A, Tortato U, Da Silva WV. Demand forecasting in food retail: A comparison between the Holt-Winters and ARIMA models. WSEAS transactions on business and economics. 2014 Jan;11(1):608-14.
- [2] Tsolacos S. Econometric modelling and forecasting of new retail development. Journal of Property Research. 1998 Jan 1;15(4):265-83.
- [3] Brooks C, Tsolacos S. Forecasting models of retail rents. Environment and Planning A. 2000 Oct;32(10):1825-39.
- [4] Hyndman RJ. Forecasting: principles and practice. OTexts; 2018. URL: <https://otexts.com/fpp2/accuracy.html>
- [5] Robert Nau, Identifying the numbers of AR or MA terms in an ARIMA model, Duke University, 2020 URL: <https://people.duke.edu/~rnau/411arim3.htm>
- [6] Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: Forecasting and control (5th ed). Hoboken, New Jersey: John Wiley & Sons.
- [7] Elasticity Based Demand Forecasting and Price Optimization for Online Retail, Chengcheng Liu, M'aty'as A. Sustik Walmart Labs, San Bruno, CA, June 17, 2021. URL: <https://arxiv.org/pdf/2106.08274>
- [8] Vagropoulos SI, Chouliaras GI, Kardakos EG, Simoglou CK, Bakirtzis AG. Comparison of SARIMAX, SARIMA, modified SARIMA and ANN-based models for short-term PV generation forecasting. In 2016 IEEE international energy conference (ENERGYCON) 2016 Apr 4 (pp. 1-6). IEEE.
- [9] Wagner C, Vinaimont T. Evaluating the wisdom of crowds. Proceedings of Issues in Information Systems. 2010 Sep;11(1):724-32.

### **Приложения**

- [1] Репозиторий проекта

URL: [https://github.com/chagrygoris/retail\\_forecasts](https://github.com/chagrygoris/retail_forecasts)