МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ
Кафедра теории вероятностей и математической статистики

ЧИГВИНЦЕВ Кирилл Александрович

ПРОГНОЗИРОВАНИЕ МАКРОЭКОНОМИЧЕСКИХ ПОКАЗАТЕЛЕЙ С ПОМОЩЬЮ МАТЕМАТИЧЕСКИХ МЕТОДОВ

Магистерская диссертация

Специальность 7-06-0533-05 «Прикладная математика и информатика»

Научный руководитель:
Меленец Юрий Витальевич
кандидат физико-математических наук,
доцент

Допущен к защите

«___»_____2025 г.

Зав. кафедрой теории вероятностей и математической статистики

_____ Харин А.Ю.

доктор физико-математических наук, профессор

Минск, 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ 6
ГЛАВА 1 Макроэкономические показатели 7
1.1 История, развитие и роль макроэкономики 7
1.2 Макроэкономические модели 8
1.3 Макроэкономические показатели и связь между ними 9
ГЛАВА 2 Математические методы прогнозирования макроэкономических показателей 17
2.1 Временные ряды 17
2.2 Тренд 18
2.3 Выявление периодических составляющих временного ряда 20
2.4 Модели авторегрессии и скользящего среднего 21
2.5 ARMA и ARIMA 24
2.6 Машинное обучение 27
2.6.1 Случайный лес 28
2.6.2 Градиентный бустинг 30
2.7 Анализ текстовых данных 31
ГЛАВА 3 Прогнозирование макроэкономических показателей 33
3.1 Прогнозирование ВВП Республики Беларусь 33
3.2 Анализ настроений на рынке акций 37
ЗАКЛЮЧЕНИЕ 40
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 41
ПРИЛОЖЕНИЕ А 42
ПРИЛОЖЕНИЕ Б 45

Общая характеристика работы

Магистерская диссертация, 47 с., 14 рис., 11 источников, 2 прил.

Ключевые слова к магистерской диссертации: МАКРОЭКОНОМИКА, ВВП, ВРЕМЕННОЙ РЯД, ТРЕНД, АВТОРЕГРЕССИЯ, СКОЛЬЗЯЩЕЕ СРЕДНЕЕ, ARMA, ARIMA, МАШИННОЕ ОБУЧЕНИЕ, СЛУЧАЙНЫЙ ЛЕС, ГРАДИЕНТНЫЙ БУСТИНГ, АНАЛИЗ НАСТРОЕНИЙ.

Объектом исследования являются показатели экономической активности и их прогноз.

Цель работы – провести исследование основных макроэкономических показателей. Построить прогнозы с помощью различных современных методов, сравнить их.

Для решения использовались такие инструменты как временные ряды, модели ARMA и ARIMA, методы машинного обучения случайного леса и градиентного бустинга, анализ текстовых данных для выявления настроений.

В результате спрогнозированы некоторые показатели с помощью различных методов, проведено их сравнение по статистическим показателям за определенный промежуток времени. В процессе исследования модели улучшались для повышения эффективности прогноза.

Агульная характарыстыка працы

Магістарская дысертацыя, 47 с., 14 мал., 11 крыніц, 2 прым.

Ключавыя словы да магістарскай дысертацыі: МАКРАЭКАНОМІКА, ВУП, ЧАСОВЫ ШЭРАГ, ТРЭНД, АЎТАРЭГРЭСІЯ, СЛІЗГАЛЬНЯЕ СЯРЭДНЯЯ, ARMA, ARIMA, МАШЫННАЕ НАВУЧАННЕ, ВЫПАДКОВЫ ЛЕС, ГРАДЫЕНТНЫ БУСТИНГ, АНАЛІЗ НАСТРОЯЎ.

Аб'ектам даследавання з'яўляюцца паказчыкі эканамічнай актыўнасці і іх прагноз.

Мэта работы-правесці даследаванне асноўных макраэканамічных паказчыкаў. Пабудаваць прагнозы з дапамогай розных сучасных метадаў, параўнаць іх.

Для вырашэння выкарыстоўваліся такія інструменты як часовыя шэрагі, мадэлі ARMA і ARIMA, метады машыннага навучання выпадковага лесу і градыентнага бустинга, аналіз тэкставых дадзеных для выяўлення настрояў.

У выніку спрагназаваны некаторыя паказчыкі з дапамогай розных метадаў, праведзена іх параўнанне па статыстычных паказчыках за пэўны прамежак часу. У працэсе даследавання мадэлі паляпшаліся для павышэння эфектыўнасці прагнозу.

General characteristics of the work

Master's thesis, 47 p., 14 fig., 11 sources, 2 appendices.

Keywords for the thesis: MACROECONOMICS, GDP, TIME SERIES, TREND, AUTOREGRESSION, MOVING AVERAGE, ARMA, ARIMA, MACHINE LEARNING, RANDOM FOREST, GRADIENT BOOSTING, SENTIMENT ANALYSIS.

The object of the study is the indicators of economic activity and their forecast.

The purpose of the work is to conduct a study of the main macroeconomic indicators. To make forecasts using various modern methods, compare them.

Tools such as time series, ARMA and ARIMA models, random forest and gradient boosting machine learning methods, and text data analysis to identify moods were used for the solution.

As a result, some indicators were predicted using various methods, and they were compared according to statistical indicators over a certain period of time. In the process of research, the models were improved to make the forecast more effective.

ВВЕДЕНИЕ

Точные прогнозы экономических показателей являются неотъемлемой частью экономического прогресса и залога успешного развития. От точности на макроуровне зависит государство, которое планирует свой бюджет на следующие годы, бизнес и инвесторы, которые оценивают благоприятность ситуации к инвестициям и расширениям производственных мощностей в стране, но также и на людей, которые являются конечным потребителем продукции и услуг. Однако, классические методы прогнозирования не учитывают влияние внешних факторов, таких как, например, настроение людей, стоимость сырья на международных рынках и политические события.

Целью данной работы является обзор основных макроэкономических показателей, разработка методов их прогнозирования с помощью таких инструментов, как анализ временных рядов, машинное обучение, анализ настроений. Задачей работы является определение наиболее точных моделей для прогноза показателей, которые могут быть использованы как инвесторами, так и рядовыми потребителями.

ГЛАВА 1 Макроэкономические показатели

1.1 История, развитие и роль макроэкономики

Экономика играет важную роль в становлении нашего общества. Без неё невозможно представить всё то, к чему мы пришли сегодня. Главная задачей экономики всегда было и есть – рациональное распределение ограниченных ресурсов для достижения максимальной степени удовлетворения наших безграничных потребностей.

Граждане, индивидуальные предприниматели, бизнес, кооперативы, государства – все являются участниками экономического процесса. У каждого есть свои потребности, свои проблемы в этом процессе. И чтобы изучать эти самые потребности и проблемы экономика выделяет в себе три основных направления: микроэкономика, макроэкономика и мировая экономика. Они отличаются рассматриваемыми субъектами и их взаимоотношениями. Мировая экономика, например, рассматривает экономический процесс на уровне государств, изучает процессы международной торговли, процессов глобализации, международного разделения труда. “Микроэкономика изучает деятельность отдельных экономических субъектов: предприятий и домашних хозяйств” [4, с.5]. Основные проблемы, которые изучает микроэкономика являются такие понятия как спрос и предложение, потребительский выбор и факторы производства. Однако со временем рассмотрение экономического процесса на уровне отдельно взятых субъектов сталкивается с серьезными проблемами, что вынуждает рассматривать экономику как нечто большее, где учитываются все участники и их взаимоотношения как единого целого механизма.

“Макроэкономика - отрасль экономической науки, изучающая поведение экономики как единого целого с точки зрения обеспечения условий устойчивого экономического роста, полной занятости ресурсов, минимизации уровня инфляции и равновесия платежного баланса” [1, с.14]. Макроэкономика рассматривает экономическую систему на уровне крупных регионов, а, зачастую, просто отдельно взятого государства.

Отсюда вытекает тот факт, что те проблемы, которые решает макроэкономика, не может на своём уровне решить микроэкономика. Основными рассматриваемыми проблемами являются:

1. Экономические циклы: колебания экономической активности, влияющие на движение экономики.

2. Общий уровень цен: инфляционные и дефляционные процессы и их влияние на экономику и потребителей.

3. Торговый баланс: влияние экспорта и импорта на национальную валюту и государственный долг.

4. Денежная масса и ключевая ставка: регулирование экономических процессов и показателей с помощью инструментов Центральных Банков.

5. Государственный бюджет: влияние фискальной политики на бизнес, взаимосвязь расходов и доходов с социальной сферой жизни.

1.2 Макроэкономические модели

В мире не существует всеобъемлющей макроэкономической модели, которая смогла бы описать все происходящие процессы и продемонстрировать связь между ними. Это банально невозможно, так как любой показатель зависит от огромного количества как внутренних, так и внешних факторов. Именно поэтому для описания и изучения процессов строятся более упрощенные модели, которые рассматривают лишь ограниченный круг переменных, и позволяют получать ответы о текущем и возможном будущем состоянии того или иного экономического фактора.

Как таковые экономические переменные в моделях делятся на две группы. Первая группа – внутренние (эндогенные) переменные. К таким переменным можно отнести инфляцию, уровень деловой активности, объемы производства, курсы валют и т.п. Вторая группа – внешние (экзогенные) переменные. К данной группе относятся такие показатели, как ключевая ставка Центрального Банка, налоговые ставки, расходы и доходы государственного бюджета.

Также существует не менее важная группировка переменных, которая рассматривает их с точки зрения момента времени. Переменные запаса определяются состоянием в текущий момент времени и могут быть лишь измерены в конкретный момент времени. Существуют также переменные потока, измерить которые можно в единицу времени. Они определяют показатели во времени. Логичным выводом является то, что именно переменные потока изменяют состояние переменных запаса. Примером того, как это работает, является модель круговых оборотов.

Если представить себе, что в экономический процесс нет вмешательства государства и (или) внешних инвестиций, то модель круговых потоков можно представить следующим образом:

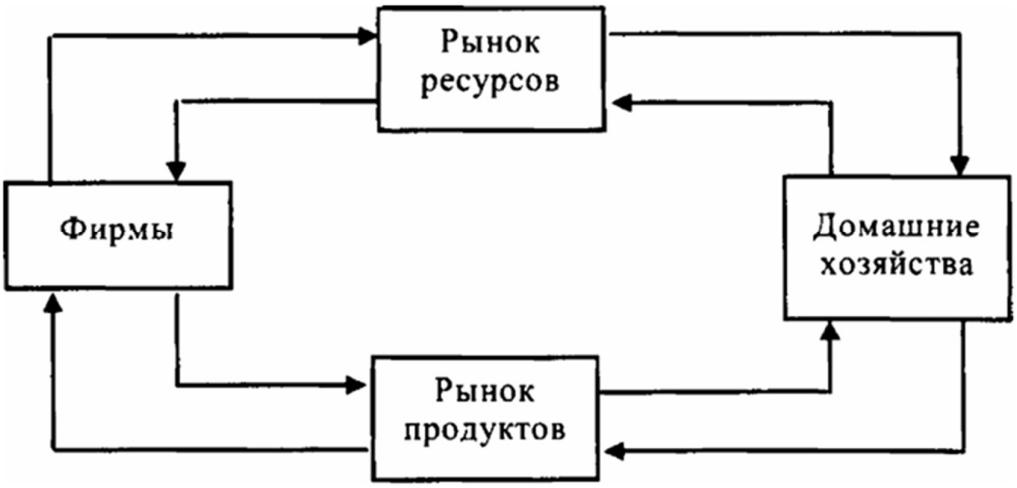


Рисунок 1.1 – Модель круговых потоков

Как видно из рисунка 1.1, модель является замкнутой. В такой системе у нас имеются два вида экономических агентов – фирмы и домашние хозяйства. И расходы одних агентов являются доходами других. И в такой замкнутой системе без внешнего вмешательства величина производства товаров и услуг фирмами равна совокупным доходам домашних хозяйств в денежном выражении.

Однако в действительности данная модель является слишком упрощенной и абстрактной. На деле она усложняется инъекциями: государственные субсидии, инвестиции, экспорт, кредиты, а также утечками: налоги, импорт, сбережения.

Всё это сводится к проблеме оптимального выбора набора рассматриваемых факторов в модели. Исходная модель круговых оборотов не может быть использована на практике, так как её упрощенность сделает исследование очень ограниченным, а выводы и вовсе окажутся неэффективными на практике. Напротив, рассмотрение всех утечек и инъекций сильно усложнит процесс исследования и сделает невозможным применение такой модели на практике. Модель должна содержать достаточное количество факторов, которые позволят исследовать процесс и выявить закономерности и тенденции, не усложняя лишними факторами и сделав применение модели простым и эффективным.

1.3 Макроэкономические показатели и связь между ними

Процесс макроэкономического анализа, как и любой анализ, использует некоторые показатели, которые отражают состояние исследуемого процесса или

объекта в тот или иной момент времени. В макроэкономике имеется широкий спектр агрегирующих показателей, которые позволяют оценить динамику и текущие положение дел в конкретных отраслях и экономике в целом.

Важнейшим показателем, который используется для оценки уровня производства в стране, является валовый внутренний продукт (ВВП). “ВВП есть рыночная стоимость всех конечных товаров и услуг, произведенных внутри страны за определенный период времени” [7, с.481].

ВВП является комплексным и сложным показателем, расчет которого учитывает множество нюансов. Существует три метода расчета ВВП:

1. по добавленной стоимости;
2. по расходам;
3. по доходам;

Расчет ВВП по добавленной стоимости также называют производственным методом. Суть данного метода заключается в суммировании стоимостей, добавленных на каждой стадии производства конечного продукта. Сумма всех добавленных стоимостей должна быть равна стоимости всех конечных товаров и услуг.

Существует и другой взгляд на расчет ВВП, который основывается на расходах участников экономического процесса. То есть каждый расход в экономике можно отнести к определенным группам. Поэтому данный показатель можно представить в виде формулы (1.1):

$$Y = C + I + G + NX, \tag{1.1}$$

где Y – есть сам показатель ВВП. Рассмотрим правую часть формулы. C отражает все потребительские расходы домашних хозяйств, I – показатель, который включает в себя траты на инвестиции, то есть покупку нового оборудования. Тут также учтены расходы домашних хозяйств по приобретению недвижимости, потому что данная трата не квалифицируется как потребительская. G представляет собой государственные расходы. И, наконец, NX отражает чистый экспорт, то есть разность между экспортом страны и её импортом.

Третий подход, основанный на расчете ВВП по доходам, заключается в том, что необходимо просуммировать доходы домашних хозяйств, прибыли организаций, проценты, ренты и прочие доходы, которые в совокупности отражают совокупный доход участников экономического процесса.

Немаловажным является исследование цен и их изменение во времени. Рост уровня цен в экономике называют инфляцией. Обратный же процесс называют дефляцией.

Расчет инфляции в первую очередь подразумевает измерение индекса потребительских цен (ИПЦ). То есть расчет стоимости среднестатистической потребительской корзины, в которую включены как материальные, так и нематериальные блага, с течением времени. Расчет уровня инфляции осуществляется по формуле (1.2):

$$\pi = \frac{CDI_1 - CDI_0}{CDI_0}, \tag{1.2}$$

где CDI_1 – ИПЦ в текущем периоде, CDI_0 – ИПЦ в прошлом периоде.

Триггером для возникновения инфляции может послужить рост совокупного спроса в экономике. То есть субъекты начинают активно тратить свои средства и скупать товары и услуги в таком темпе, который не может обеспечить текущий уровень предложения. Причинами такого вида инфляции могут послужить дешевые кредиты, которые в моменте выдаются потребителям и бизнесу, а также номинальный рост доходов домашних хозяйств. С другой стороны, и дисбаланс со стороны предложения тоже может вызывать инфляцию.

Ещё одним фактором возникновения инфляции является рост издержек производства. В такой ситуации производства тратят больше денежных ресурсов на изготовление одной единицы продукции, что в свою очередь ведет к снижению доходов и сокращению объемов производства. Причинами такой инфляции может быть:

1. рост стоимости сырья и материалов, которые используются в процессе производства;
2. введение налогов и пошлин;
3. несоразмерный рост реальных доходов и производительности труда;

Отдельным фактором выделяют так называемые инфляционные ожидания. Под этим понятием подразумевают ожидаемые, планируемые уровни инфляции, которые влияют на субъектов экономических отношений в разрезе их будущих трат и сбережений. Исследования инфляционных ожиданий в Беларуси производит Национальный Банк РБ среди 2000 выбранных респондентов на основании анкеты, где содержится три блока вопросов: цены, покупки и сбережения. На рисунках 1.2, 1.3, 1.4 представлены результаты такого исследования, проведенного в декабре 2021 года.

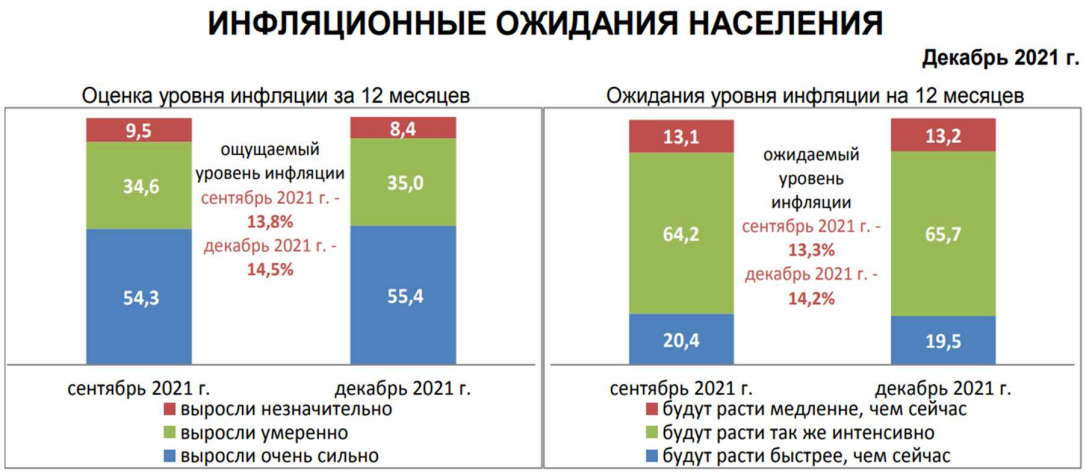


Рисунок 1.2 – Исследования цен

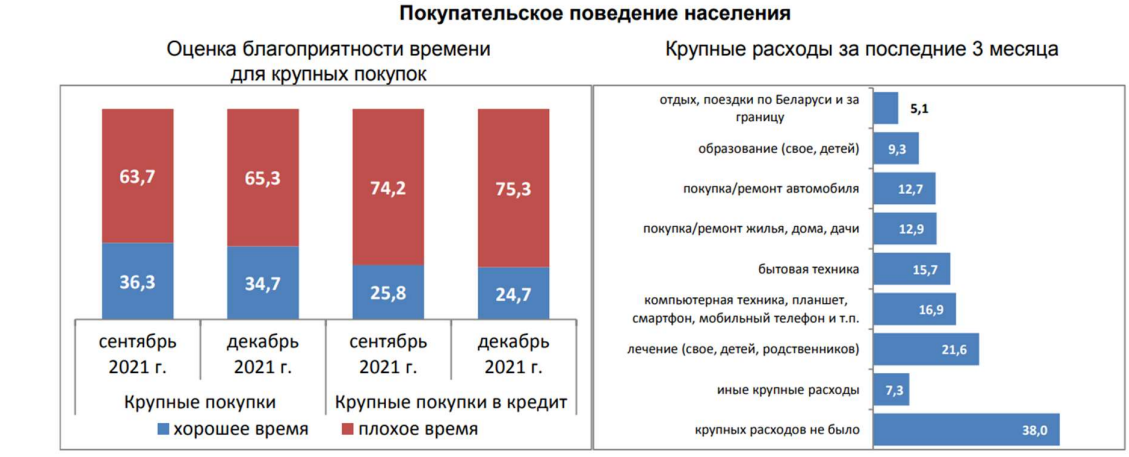


Рисунок 1.3 – Исследование покупательского поведения

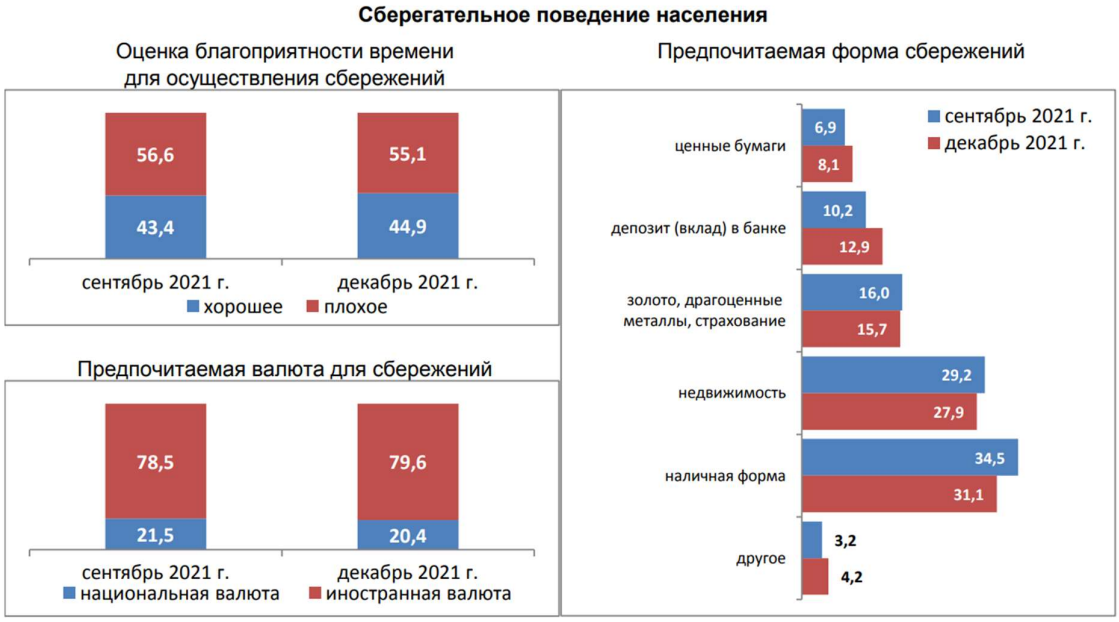


Рисунок 1.4 – Исследования сберегательного поведения

Инфляция может нести риски уменьшения реальных доходов домашних хозяйств. В случае ожидаемой инфляции субъект может предпринять действия по сбережению и инвестированию, чтобы уберечь свои накопления от обесценивания. Номинальный доход может быть скорректирован с учётом ожидаемой инфляции с помощью уравнения Фишера (1.3):

$$r = i - \pi^e, \tag{1.3}$$

где r, i – реальная и номинальная ставка процента соответственно, π^e – ожидаемый уровень инфляции. Однако данная формула справедлива только при умеренной инфляции, то есть значений до 10%. При более высокой инфляции формула (1.3) имеет вид (1.4):

$$r = \frac{i - \pi^e}{1 + \pi^e}. \tag{1.4}$$

Важным показателем, на который всегда обращают внимание при оценке состояния экономики, является занятость населения. Под занятостью населения понимается вовлеченность трудоспособного слоя населения, или иначе экономически активного, в производство и потребление товаров и услуг.

“Безработица – это превышение предложения труда над спросом на труд” [9, с.78]. Существует несколько видов безработицы. Фрикционная безработица

понимается как отсутствие работы на данный момент, но продолжающийся её поиск на рынке труда. То есть это те люди, которые решили на добровольной основе покинуть текущее место работы, однако ещё не успели найти новое место. Этот тип безработицы характеризуется краткосрочным эффектом и добровольностью трудового ресурса в таком решении. А вот если у работодателя произошли изменения в процессе производства товаров и услуг и появились специалисты, которые больше не требуются, то такой тип безработицы называют структурным. Зачастую такие изменения происходят в следствии внедрения достижений научно-технического прогресса в производство. Структурная безработица носит уже недобровольный характер и эффект от неё является долгосрочным, так как требуется немало времени на переквалификацию освободившегося трудового ресурса.

Совокупность фрикционной и структурной безработицы образуют естественную безработицу. А разница между фактической безработицей в стране и естественной называют циклической. Связано это с циклическими колебаниями рыночной экономики: в момент спада данный вид безработицы дополняет существующие, а в момент подъема – отсутствует.

Уровень безработицы u оценивает текущее состояние занятости ресурсов и может быть рассчитан по формуле (1.5):

$$u = \frac{U}{L}, \tag{1.5}$$

где U – численность безработных, L – численность рабочей силы.

В процессе развития макроэкономики всё больше экономистов уделяло внимание поиску зависимостей между различными показателями и факторами. И, конечно, фактический уровень занятости населения и объемы производства вызывали интерес, так как их взаимосвязь логична. Чем больше рабочей силы вовлечено в производственный процесс, тем больше в экономике производится товаров и услуг, что соответственно и является ростом ВВП. Эту зависимость описывает закон Оукена, который говорит о том, что “превышение фактического уровня безработицы на 1% над естественным уровнем приводит к уменьшению фактического реального ВВП на 2,5-3% по сравнению с потенциально возможным ВВП (т.е. ВВП при полной занятости)” [10, с.27]. На графике эту зависимость можно представить как изображено на рисунке 1.5:

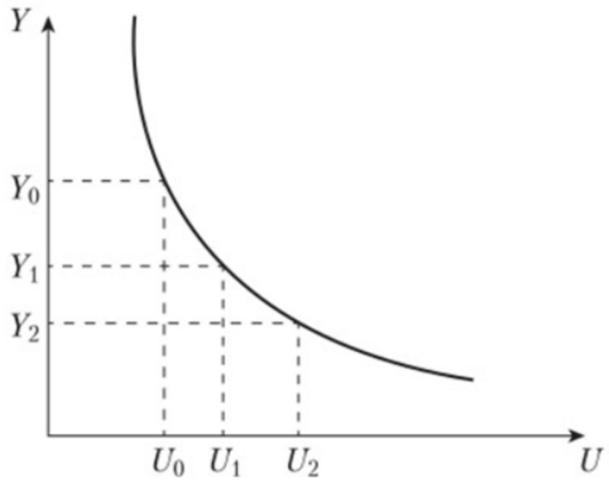


Рисунок 1.5 – График зависимости ВВП и безработицы

При снижении уровня занятости с U_0 до U_1, снижается и объем производства с базового Y_0 до Y_1, что говорит об обратной зависимости между уровнем безработицы и объемом производства.

Также этот закон можно представить в виде следующей формулы (1.6):

$$\frac{Y - Y^*}{Y^*} = -\beta(u - u^*), \tag{1.6}$$

где Y и Y^* – фактический и потенциальный объемы производства соответственно, u и u^* – фактический и естественный уровень безработицы, β – корректирующий коэффициент, который отражает чувствительность ВВП к циклической безработице.

Существует и зависимость между инфляцией и безработицей. Чем ниже уровень безработицы, тем больше ресурсов задействовано в цикле производства. Значит всё меньше свободных ресурсов остается на рынке, что вызывает инфляцию спроса. И в такой ситуации чтобы снизить инфляцию необходимо увеличить уровень безработицы. То есть в краткосрочном периоде существует обратная зависимость между инфляцией и безработицей. Кривая Филлипса, которая изображена на рисунке 1.6, отражает эту зависимость

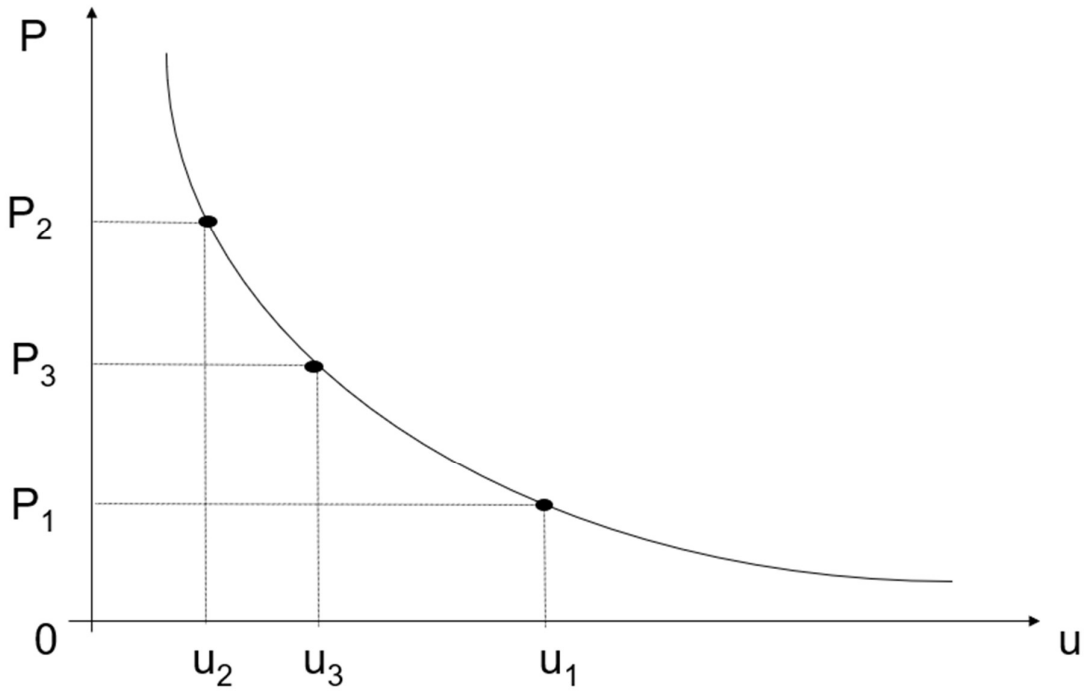


Рисунок 1.6 – Кривая Филлипса

Соответственно, любая точка на кривой Филлипса является компромиссом между уровнем безработицы и инфляции. Формально этот компромисс можно представить в следующем виде (1.7):

$$\pi = \pi^e + f\left(\frac{Y - Y^*}{Y^*}\right) + \varepsilon, \tag{1.7}$$

где π и π^e – фактический и ожидаемый уровни инфляции, $f\left(\frac{Y-Y^*}{Y^*}\right)$ – инфляция спроса, ε – инфляция издержек. Однако используя формулу (1.6), получим (1.8):

$$\pi = \pi^e + \gamma(-\beta(u - u^*)) + \varepsilon, \tag{1.8}$$

где инфляцию спроса мы заменяем фактический и естественный уровень безработицы с корректирующем коэффициентом γ.

ГЛАВА 2 Математические методы прогнозирования макроэкономических показателей

2.1 Временные ряды

Макроэкономические показатели представляют собой некоторую числовую оценку на определенный момент времени. “Ряд наблюдений $y(t_1), y(t_2), \dots, y(t_n)$ анализируемой случайной величины $\xi(t)$, произведенных в последовательные моменты времени $t_1, t_2, \dots, t_n$, называется временным рядом, то есть временной ряд есть упорядоченная во времени последовательность наблюдений, характеризующих уровни развития изучаемого явления в последовательные моменты или периоды времени” [2, с.7].

В данном определении величины $y(t_1), y(t_2), \dots, y(t_n)$ называются уровнями ряда, а $t_1, t_2, \dots, t_n$ – временными метками или моментами. Исследование временных рядов направлено на выявление закономерностей в развитии уровней ряда с дальнейшим построением модели, способной учесть особенности развития, для прогнозирования будущих значений.

В зависимости от различных характеристик существует некоторая классификация временных рядов:

1. Если наблюдения $t_1, t_2, \dots, t_n$ сделаны через равные промежутки времени, то такой временной ряд называют равномерным. В противном случае – неравномерным.

2. Если исследуется лишь одна характеристика временного ряда, то тогда его называют одномерным временным рядом. Если же параметров два и более, то многомерным.

3. Если ряд наблюдений описывается формулами и является предсказуемым, то такой временной ряд называют детерминированным. В противном случае, что на практике встречается чаще, временной ряд имеет вероятностную компоненту.

4. Если временной ряд сохраняет с течением времени свои статистические свойства, то есть уровни ряда сохраняют не меняют совместное распределение, то такой ряд называется стационарным. Если же ряд обладает такими свойствами как тренд, сезонность, шум, то такой ряд уже называют нестационарным.

Стационарность является ключевым критерием, который влияет на ход проведения анализа временного ряда.

Рассмотрим временной ряд $y(t)$. В любую временную метку его можно представить по формуле (2.1):

$$y(t_i) = f(t_i) + \varphi(t_i) + \psi(t_i) + \varepsilon(t_i), \qquad (2.1)$$

где функция f соответствует функции тренда, φ – сезонная функция, ψ – циклическая функция, ε – нерегулярная компонента, представляющая собой случайную величину, t_i – i-ая временная метка.

Мультипликативная модель (2.1) имеет вид (2.2):

$$y(t_i) = f(t_i)\varphi(t_i)\psi(t_i) + \varepsilon(t_i) \tag{2.2}$$

Первых три слагаемых (2.2) называются систематическими компонентами. Функция тренда f отражает долгосрочную установившуюся тенденцию развития уровней ряда. Например, рост показателя на рассматриваемом промежутке. Функция φ, которая описывает сезонность, демонстрирует влияние факторов, периодичность которых составляет менее 1 года. В то время как функция цикличности ψ описывает более длительные периоды, то есть 1 год и более. Выявление систематических компонент может происходить с помощью визуализации временного ряда, так и с помощью различных статистических методов. Компонента ε описывает случайные процессы разнообразной структуры.

Сам по себе анализ временных рядов представляет собой выделение этих систематических компонент и описанию случайной части.

2.2 Тренд

Важным фактором при построении прогнозных моделей является наличие тренда в исследуемом процессе, то есть наличие тенденции развития или стагнации с изменением времени. “Тренд является простейшей регрессионной моделью, в которой рассматриваемый показатель является зависимой переменной, а время – независимой” [3, с.49].

Любая трендовая модель может быть представлена в виде (2.3):

$$\widehat{y_{T+h}} = f(T + h), \tag{2.3}$$

где h - период прогнозирования, T – время или номер последнего известного ряда данных, f – функция тренда, выбор которой зависит от набора данных, $\widehat{y_{T+h}}$ – прогнозное значение показателя на момент времени $T + h$. То есть логика трендового прогнозирования заключается в том, что значение показателя в

будущем ищется с помощью выбора нужного нам времени или номера наблюдения в будущем и подстановкой в готовую модель.

В зависимости от типа изменения показателя во времени выделяют следующие виды тренды:

1. При равномерном изменении показателя – линейный тренд, определяющийся по формуле (2.4):

$$\widehat{y_t} = a_0 + a_1 t. \tag{2.4}$$

2. При квадратичной зависимости показателя – параболический тренд, который представим в виде (2.5):

$$\widehat{y_t} = a_0 + a_1 t + a_2 t^2. \tag{2.5}$$

3. При взрывном росте показателя или замедленной стагнации выделяют показательный тренд (2.6):

$$\widehat{y_t} = a_0 a_1^t. \tag{2.6}$$

Часто используют замену $b_1 = lna_1$ и приходят к виду (2.7) показательного тренда:

$$\widehat{y_t} = a_0 e^{b_1 t}. \tag{2.7}$$

4. При плавном росте показателя с выходом на некоторый стабильный уровень выделяют гиперболический тренд (2.8):

$$\widehat{y_t} = a_0 + {a_1}/{t}. \tag{2.8}$$

5. Если зависимость показателя носит степенной характер, то такой тренд определяется по формуле (2.9):

$$\widehat{y_t} = a_0 + a_1 t^{a_2}. \tag{2.9}$$

2.3 Выявление периодических составляющих временного ряда

“Любой ряд динамики можно представить как конечную сумму некоторых гармоник” [6, с.40]. Среди всех гармоник выделяют в анализ лишь те, кто существенно влияет на дисперсию процесса. Разложение ряда в гармоники называется спектральным анализом, а основным инструментом для этого являются преобразования Фурье.

Рассмотрим ряд, определяющийся по формуле (2.10):

$$x_n = x(t_n), \qquad t_n = n\Delta t, \qquad n = 0,1,\dots,N-1 \tag{2.10}$$

с нулевым математическим ожиданием и временем реализации T. Временной параметр представлен дискретно, то есть T можно представить в виде $\Delta t * N$, поэтому чаще всего используют дискретное преобразование Фурье.

В таком случае, частота вычисляется по формуле (2.11):

$$f_k = {}^{k}/_{\Delta tN}\ , \tag{2.11}$$

где k – номера гармоник. Тогда коэффициенты ряда Фурье имеют вид (2.12):

$$X(f_k) = \sum_{n=0}^{N-1} x_n e^{-i\frac{2\pi kn}{N}}, \qquad k = 0,1,\dots,N-1. \tag{2.12}$$

Оценивая спектральную плотность, можем выявить связь между x_i и гармонической составляющей ряда с помощью формулы (2.13):

$$S(f_k) = \frac{1}{\Delta tN}|X_i(f_k)|^2. \tag{2.13}$$

После удаления трендовой составляющей временного ряда и проведении спектрального анализа по выявлению значимых гармоник, проводится гармонический анализ. Основой является тот факт, что любой временной ряд можно представить в виде конечного ряда Фурье. Для каждой точки такого ряда имеет место выражение (2.14):

$$y_t = f_t + \sum \left(a_k \cos\left(kt\frac{2\pi}{n}\right) + b_k \sin\left(kt\frac{2\pi}{n}\right)\right), \tag{2.14}$$

где y_t – фактический уровень ряда в момент времени t, f_t – выровненный уровень ряда в тот же момент времени t, a_k и b_k коэффициенты ряда. Они позволяют оценивать масштаб отклонения от общий тенденции относительно начальной точки. Эти коэффициенты определяются согласно формулам (2.15) – (2.17):

$$\varepsilon_t = y_t - f_t. \tag{2.15}$$

$$a_k = \frac{2}{n}\sum_{t=1}^{n} \varepsilon_t \cos\left(\frac{2\pi}{n} kt\right). \tag{2.16}$$

$$b_k = \frac{2}{n}\sum_{t=1}^{n} \varepsilon_t \sin\left(\frac{2\pi}{n} kt\right). \tag{2.17}$$

2.4 Модели авторегрессии и скользящего среднего

В исследовании экономических процессов можно часто встретить тезисы о том, что текущее или будущее состояние процесса зависит от его прошлого состояния. Модели, где значение в следующий момент времени зависит от значения в текущий момент времени, называют моделями авторегрессии. Формально это выражается так: значение исследуемого показателя y_t в момент времени t имеет влияние на значение этого же показателя в моменты времени $t+1, t+2, \dots, t+h$.

Модель авторегрессии обозначается как $AR(p)$, в которой параметр p означает, от какого количество наблюдений назад зависит текущее значение модели. Саму модель можно представить в следующем виде:

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_p y_{t-p} + c + \varepsilon_t. \tag{2.18}$$

В модели (2.18) a_i – это коэффициенты, поиск которых можно осуществить с помощью метода наименьших квадратов, их же значимость можно проверить стандартными методами. c – константа, ε_t – ошибка модели. Предполагается, что ошибка имеет нулевое матожидание и постоянную дисперсию.

Рассмотрим лаговый оператор, чтобы упростить запись исходной модели. Данный оператор представляется в следующем виде:

$$B^p y_t = y_{t-p}. \tag{2.19}$$

Используя лаговый оператор (2.19) и оставив по одну сторону знака равенства только константу и ошибку, получим:

$$y_t\left(1 - a_1 B - a_2 B^2 - \cdots - a_p B^p\right) = c + \varepsilon_t. \tag{2.20}$$

Форма записи (2.20) является общепринятой для авторегрессии порядка p. Выражение в скобках можно представить как функцию от оператора сдвига и обозначить $\varphi_p(B)$. Тогда исходная форма записи модели упроститься до выражения (2.21):

$$\varphi_p(B) y_t = c + \varepsilon_t. \tag{2.21}$$

Также стоит отметить, что рассмотрение константы в модели авторегрессии не является обязательным. Она меняет уровень самого ряда, но не имеет влияние на смысл.

Рассмотрим авторегрессию первого порядка $AR(1)$. Такую модель именуют Марковским процессом, и она имеет следующий вид:

$$y_t = a_1 y_{t-1} + c + \varepsilon_t. \tag{2.22}$$

Исходя из модели (2.22), прогноз на один шаг вперед представляется (2.23):

$$\widehat{y_{T+1}} = a_1 y_T + c. \tag{2.23}$$

Аналогичной подстановкой получаем значение на два шага вперед (2.24)

$$\widehat{y_{T+2}} = a_1 y_{T+1} + c = a_1(a_1 y_T + c) + \mathrm{c} = \mathrm{c}(1 + a_1) + a_1^2 y_T. \tag{2.24}$$

Таким образом можно на итеративной основе можем получить формулу для расчета значения на h шагов вперед:

$$\widehat{y_{T+h}} = a_1^h y_T + c\left(1 + a_1 + a_1^2 + \cdots + a_1^{h-1}\right). \quad (2.25)$$

Несложно заметить, что в (2.25) выражение в скобках представляет собой ни что иное, как геометрическую прогрессию. Посчитав сумму этой прогрессии, получим выражение для расчета показателя через h наблюдений после последнего известного для модели авторегрессии первого порядка $AR(1)$:

$$\widehat{y_{T+h}} = \mathrm{c}\frac{a_1^h - 1}{a_1 - 1} + a_1^h y_T. \quad (2.26)$$

Будущие значения, исходя из формулы (2.26), зависят в первую очередь от коэффициента a_1, причем зависимость это имеет показательный характер. В случае, если коэффициент $a_1 = 1$, то зависимость уже будет идти от случайной ошибки ε_t и такой процесс называется процессом случайного блуждания.

В процессе развития самой авторегрессии получило место идея о том, что текущее значение может и зависеть от ошибок при прошлых наблюдениях, которые распределены нормально. Так и появилась модель скользящего среднего, обозначаемая $MA(q)$. Общий вид такой модели имеет следующий вид:

$$y_t = c_1\varepsilon_{t-1} + c_2\varepsilon_{t-2} + \cdots + c_q\varepsilon_{t-q} + \varepsilon_t. \quad (2.27)$$

Используя уже ранее введенный оператор лага, запись (2.27) упроститься:

$$y_t = \varepsilon_t\left(1 + c_1 B + c_2 B^2 + \cdots + c_q B^q\right). \quad (2.28)$$

Введем функцию, которая описывает выражение в скобках и является оператором сдвига и получим вид (2.29) исходной модели $MA(q)$:

$$y_t = \varepsilon_t \theta_q(B). \quad (2.29)$$

Для расчета коэффициентов c_i здесь уже применяются численные методы. Также стоит отметить, что сумма коэффициентов c_i необязательно должна быть равна 1.

В ситуации с скользящим средним также требуется уметь определять значение через h наблюдений. Для этого рассчитывается следующее значение от последнего известного и делается предположение, что они равны. То есть $y_{T+1} = \widehat{y_{T+1}}$. А так как разность фактического и прогнозного значения ряда и есть

ошибка, то выходит, что ε_t. А это значит, что прогноз представляет собой прямую. Тогда значение уровня ряда через промежуток времени h для модели $MA(q)$ получается по формуле (2.30):

$$y_{T+1} = \widehat{y_{T+h}}. \tag{2.30}$$

Однако перед началом использования модели скользящего среднего необходимы начальные значения ε_t. Их задают с помощью обратного прогноза. Для этого делается предположение, что значение на момент времени $t-1$ и $t+1$ одинаково коррелируют со значением в момент времени t, что позволит запустить процесс поиска с конца ряда данных. Исходя из формулировки исходной модели (2.27) значение ошибок вычисляется по формуле (2.31):

$$\varepsilon_t = y_t - c_1\varepsilon_{t-1} - c_2\varepsilon_{t-2} - \cdots - c_q\varepsilon_{t-q}. \tag{2.31}$$

Расчет таким образом ошибок приведет к наличию стартовых q ошибок, которые уже по-другому влияют на расчетные уровни ряда, поэтому итеративно происходит пересчет начальных ошибок и коэффициентов, пока не удастся получить устойчивые оценки.

2.5 ARMA и ARIMA

В модели авторегрессии выразим y_t, опустив при этом значение константы. Используя укороченную форму записи (2.21), получим (2.32):

$$y_t = \frac{\varepsilon_t}{\varphi_p(B)}. \tag{2.32}$$

Если выражать в формуле (2.29) значения y_t через саму же формулу, то получим следующее:

$$y_t = \theta(B)\varepsilon_t, \tag{2.33}$$

то есть получился бесконечный ряд ошибок с коэффициентами авторегрессии. Таким образом, модель конечного порядка авторегрессии соответствует модели бесконечного порядка скользящего среднего. Справедливо и обратное утверждение, что говорит о связи модели $AR(p)$ и $\mathrm{M}A(q)$. Вместе эти модели образуют модель $ARMA(p,q)$. В общем виде она имеет вид (2.34):

$$y_t = c + a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_p y_{t-p} + c_1 \varepsilon_{t-1} + c_2 \varepsilon_{t-2} + \cdots + c_q \varepsilon_{t-q} + \varepsilon_t. \quad (2.34)$$

Упрощенная форма записи с помощью лаговых операторов имеет вид (2.35):

$$\varphi_p(B) y_t = \theta_q(B) \varepsilon_t + c. \quad (2.35)$$

Важным условием применение модели $ARMA(p, q)$ является стационарность исследуемого процесса. Существует универсальный способ проверки данного факта для моделей $AR(p)$. Его идея заключается в нахождение корней характеристического уравнения. Представим модель в виде

$$y_t\left(1 - a_1 B - a_2 B^2 - \cdots - a_p B^p\right) = \varepsilon_t. \quad (2.36)$$

Тогда характеристическое уравнение для (2.36) имеет вид (2.37):

$$1 - a_1 X - a_2 X^2 - \cdots - a_p X^p = 0. \quad (2.37)$$

Имея коэффициенты и решив это уравнение, получим p корней. Они могут быть как действительными, так и комплексными. Сам критерий стационарности можно выразить следующим образом:

$$|X| > 1, \quad (2.38)$$

то есть модуль числа должен не лежать в единичной окружности. В противном случае, процесс будет нестационарным.

Аналогичный критерий имеется и для понятия обратимости. Формально это означает, что “если модель скользящего среднего порядка q можно представить как модель авторегрессии бесконечного порядка, то процесс считается обратимым”[5, с.71]. И данное условие выполняется, если корни характеристического уравнения $MA(q)$ лежат в единичной окружности. Модели $MA(q)$ в виде (2.39):

$$y_t = \varepsilon_t\left(1 + c_1 B + c_2 B^2 + \cdots + c_q B^q\right) \quad (2.39)$$

соответствует характеристическое уравнение (2.40):

$$1 + c_1 X + c_2 X^2 + \cdots + c_q X^q = 0 \quad (2.40)$$

и условие обратимости выражается аналогично (2.38)

Зачастую рассматриваемые экономические процессы имеют тренды или сезонность. Все подобные явления говорят нам о том, что рассматриваемый процесс, вероятно, будет нестационарным. А значит применение вышеописанных методов может привести к неточным результатам. В таких ситуациях следует избавляться от нестационарности.

Если рассматриваемый ряд нестационарен, то переходят к рассмотрению ряда его разностей, то есть рассматриваются разности вида (2.41)

$$\Delta^1 y_t = y_t - y_{t-1}, \quad (2.41)$$

где Δ^1 – означает разность первого порядка. Если после применения разности тренд всё ещё сохранился и нестационарность присутствует, тогда берут ещё одну разность, то есть уже будет разность второго порядка вида (2.42):

$$\Delta^2 y_t = \Delta^1 y_t - \Delta^1 y_{t-1} = y_t - y_{t-1} - y_{t-1} + y_{t-2} = y_t - 2y_{t-1} + y_{t-2}. \quad (2.42)$$

Для формальной записи рассмотрим лаговый оператор и преобразуем выражение (2.41) следующим образом:

$$\Delta^1 y_t = (1 - B) y_t. \quad (2.43)$$

Аналогично, разность второго порядка имеет вид (2.44):

$$\Delta^2 y_t = (1 - B)^2 y_t. \quad (2.44)$$

И, вообще говоря, разность любого порядка представляется в виде (2.45):

$$\Delta^d y_t = (1 - B)^d y_t. \quad (2.45)$$

На практике не используют обычности разности выше второй. Так как уже она приводит практически все нестационарные процессы к стационарным.

Модель авторегрессии со скользящим средним, построенная на разностях носит название $ARIMA(p, d, q)$, где новый параметр d отвечает за порядок разностей. Саму модель можно представить в виде:

$$\varphi_p(B)(1 - B)^d y_t = \theta_q(B)\varepsilon_t + c. \quad (2.46)$$

Константу c из модели (2.46) зачастую убирают, если параметр d положительный. Потому что при построении разностей она образует линейный тренд. Хотя если не убирать константу, то исходную модель также называют моделью с дрифтом.

Многообразие описанных моделей позволяет для каждого временного ряда подобрать свой метод исследования, что является несомненным плюсом использование моделей $ARIMA$. Также большим достоинством является факт наличия четкой и обоснованной статистической базы, что позволяет легко обосновать полученные данной моделью результаты.

Однако данные модели не лишены недостатков. Они сложны в реализации и используют достаточное количество ресурсов. При этом само построение требует наличия опыта и «профессионального взгляда» для получения более точных результатов. Ещё одним важным недостатком является не способность моделей к восприятию новых данных. В случае их появления – придется переоценивать модель. Также можно отметить, что для построения хорошей модели требуется не менее 40 наблюдений, чего иногда не может быть в наличии у аналитика.

2.6 Машинное обучение

В процессе решения аналитических задач возникают сложности с построением и обработкой моделей, которые учитывают в себе большое количество факторов. Также существенной проблемой является факт обновления и изменений ряда данных, которые также должны быть во внимании. Однако с такой задачей в современном мире справляется искусственный интеллект. Специализированные модели поддаются обучению и умеют принимать решения как реальные люди. Одним из важнейших направлений в аналитике данных являются модели машинного обучения.

“Машинное обучение (ML) – это подмножество методов искусственного интеллекта, которое позволяет компьютерным системам учиться на предыдущем опыте (то есть на наблюдениях за данными) и улучшать свое поведение для выполнения определенной задачи. Методы ML включают методы опорных

векторов (SVM), деревья решений, байесовское обучение, кластеризацию k-средних, изучение правил ассоциации, регрессию, нейронные сети и многое другое” [8, с.17].

Когда классические подходы к решению задач, такие как кластеризация, регрессия и классификация не позволяют найти решения, то прибегают к методам ML. В методах машинного обучения можно выделить 5 групп методов: обучение без учителя, обучение с учителем, полууправляемое обучение, глубокое обучение и обучение с подкреплением. Данные методы различаются, в первую очередь, по цели использования и имеющемуся набору данных. Например, если у нас имеется ряд данных x и соответствующий ему ряд со значениями y, то используют методы обучения с учителем. Если же ряд данных значений y отсутствует, то модель строить лучше, основываясь на обучение без учителя.

Реализуются модели машинного обучения в основном на языке программирования Python, используя уже написанные библиотеки, где реализованы вышеописанные методы.

2.6.1 Случайный лес

Одним из распространенных методов, применяемых в машинном обучении, является случайный лес. Данный метод применим как к задачам классификации, так и задачам регрессии. По факту, случайный лес представляет собой множество деревьев решений. Принцип работы метода следующий: всю выборку делят на некоторые подмножества, размерность которых не должна превышать размерность исходного множества, причем разделение происходит случайным образом, то есть элементы одного дерева могут пересекаться с элементами других деревьев. Данный процесс представлен на рисунке 2.1.

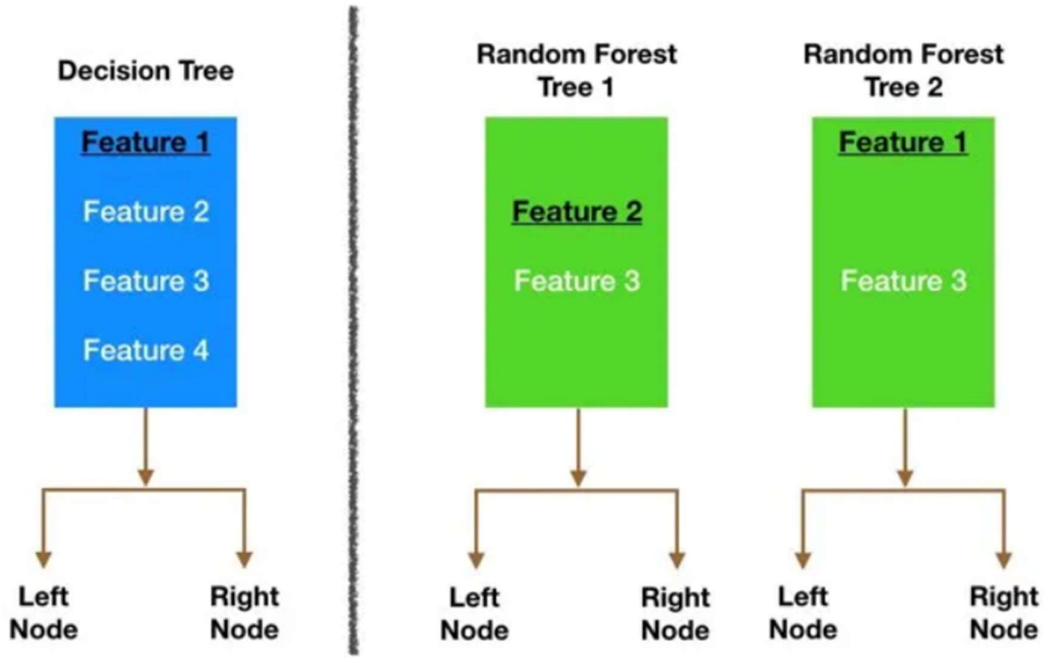


Рисунок 2.1 – Разделение выборки на случайные подмножества

И далее для каждого из деревьев реализуется алгоритм дерева решений. Результаты для каждого из деревьев получаются независимыми друг от друга и в дальнейшем усредняются. Признаки в вершинах деревьев также выбираются случайным образом, причем совокупность признаком для последующий вершины учитывает признаки предыдущей. Формально такой процесс можно представить в виде формулы (2.47):

$$a(x) = \frac{1}{N}\sum b_i(x)\,, \tag{2.47}$$

где N – количество деревьев, $b_i(x)$ – i-тое дерево, $a(x)$ – ансамбль решений, x – сгенерированная выборка.

К преимуществам использования данного метода можно отнести устойчивость к переобучению, что обусловлено самой структурой метода. Также данный метод позволяет работать с различными типами данных, будь то категориальные или количественные, а также эффективно обрабатывать пропуски данных. Но существенным недостатком можно назвать большой объем занимаемой памяти. Так для N деревьев потребуется $O(N)$ памяти для их хранения.

2.6.2 Градиентный бустинг

Теоретической основой для бустинга является допущение о том, что из совокупности слабых моделей можно получить одну сильную. Слабой моделью в данной ситуации называется такая модель, точность который немногим выше точности случайного угадывания значений. “Совокупность таких слабых моделей должна быть линейная комбинация с набором некоторых весов, которые при каждой итерации переоцениваются” [11, с.237].

Пусть у нас имеется набор данных x и набор решающих значений y. Для такого набора данных у нас имеется модель $a(x)$. Тогда построим модель $b(x)$ таким образом, чтобы:

$$y_i = a(x_i) + b(x_i). \tag{2.48}$$

В модели (2.51) разность правильных ответов и результатов модели $a(x)$ называется невязка и обозначается ε. Вообще говоря, алгоритм $b(x)$ выступает алгоритмом, который корректирует результаты имеющегося под правильные ответы. Пусть $L(y, a)$ – функция ошибки, тогда задача поиска алгоритма $b(x)$ сводится к следующей задаче минимизации (2.49)

$$\sum L(y_i, a(x_i) + b(x_i)) \to min\,. \tag{2.49}$$

Так как функция достигает своего минимума при движении в противоположную сторону движения градиенту, то

$$b_i = -L'\big(y_i, a(x_i)\big). \tag{2.50}$$

Таким образом, обучение алгоритма $b(x)$ должно происходить на выборке $(x_i, -L'(y_i, a(x_i))$.

Достоинства использования данной модели заключаются в точности и гибкости настроек. Однако, такой метод склонен к переобучению, что можно исправлять подбором гиперпараметров, и из-за последовательного подхода градиентный бустинг отрабатывает медленнее, чем тот же случайный лес.

2.7 Анализ текстовых данных

В сфере экономике немаловажное место занимает поведение потребителя. Именно от совокупности возможностей и потребностей потребителей выстраивается рынок. Помимо количественных описаний этих параметров, таких как, например, зарплаты, стоимость аренды жилья, есть ещё и другие факторы, которые не могут быть выражены численно и отражены в каких-либо моделях. Некоторые факты выражаются через восприятие человеком ситуации на рынке. Ярким примером такой ситуации может служить инфляционное ожидание. Тот факт, что люди ждут роста цен и снижение реальных доходов может быть отражен лишь в словах. Или новости, которые публикуются СМИ, имеют только текстовый вид, а их влияние на рынки и цены имеет существенный вес. Поэтому помимо цифр в экономике немаловажно уметь и обрабатывать текст. И благодаря развитию искусственного интеллекта это стало возможным.

Область искусственного интеллекта, цель который выстроить систему для понимания смысла человеческой речи, называется Natural Language Processing или сокращенно NLP.

Различные методы и подходы в NLP позволяют разбирать тексты, выделять в них настроения и самые часто встречаемые слова, а также генерировать тексты и распознавать голос.

Первым этапом обработки текста является его токенизация. Под этим этапом подразумевается разбиения текста на какие-либо его части, то есть токены. Зачастую, этими частями являются слова, а разделение происходит по пробелам и знакам препинания. Наиболее часто используемой библиотекой в Python для таких целей выступает пакет NLTK. А вот популярная модель от Google по распознаванию контекста BERT разбивает сами слова на некоторые составные части и эти части считает токенами.

Далее необходимо очистить текст от стоп-слов. К ним относятся предлоги, междометия, союзы и т.п. Для их очистки можно использовать готовые словари стоп-слов в Python.

Следующим этапом является стэмминг слова, то есть их упрощение. В любом языке слова имеют различные формы в зависимости от рода, падежа, склонения. И анализ всех этих форм не является целесообразным, поэтому слова нужно упростить до корня, убирая суффиксы, префиксы, приставки и подобное. Более глубокий морфологический анализ чем вышеописанный позволяет делать лемматизация, но их суть одинакова.

И далее уже по позитивной или негативной окраске слова, которая может быть определена в рамках пакетных решений, так и предоставлением входных

данных, строится словарь частотности и по совокупности слов определяется тональность текста – позитивная, негативная и нейтральная.

Наиболее часто встречающимся способом визуализации частотности встречаемости слов в тексте является в виде облака слов. Для этого используется библиотека wordcloud. На вход подается лишь одна строка, поэтому все токены текста необходимо соединить в строку через пробел. На выходе же мы получим визуализированный анализ частотности как показано на рисунке 2.2.

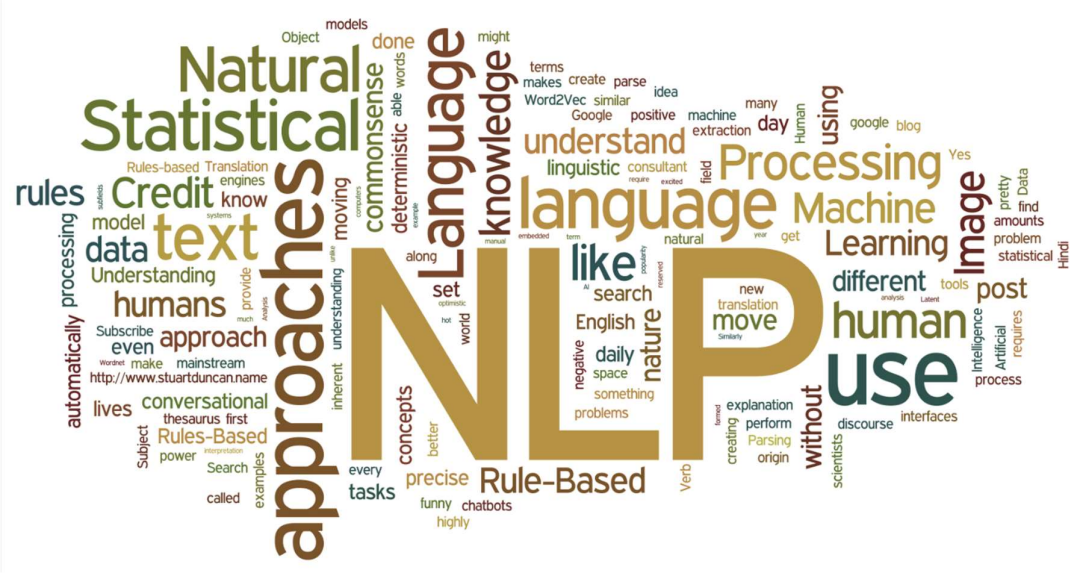


Рисунок 2.2 – Облако слов

ГЛАВА 3 Прогнозирование макроэкономических показателей

3.1 Прогнозирование ВВП Республики Беларусь

В качестве исходных данных взят показатель ВВП РБ за период с 1990 по 2022 год включительно в долларовом эквиваленте. Динамика развития данного показателя представлена на рисунке 3.1:

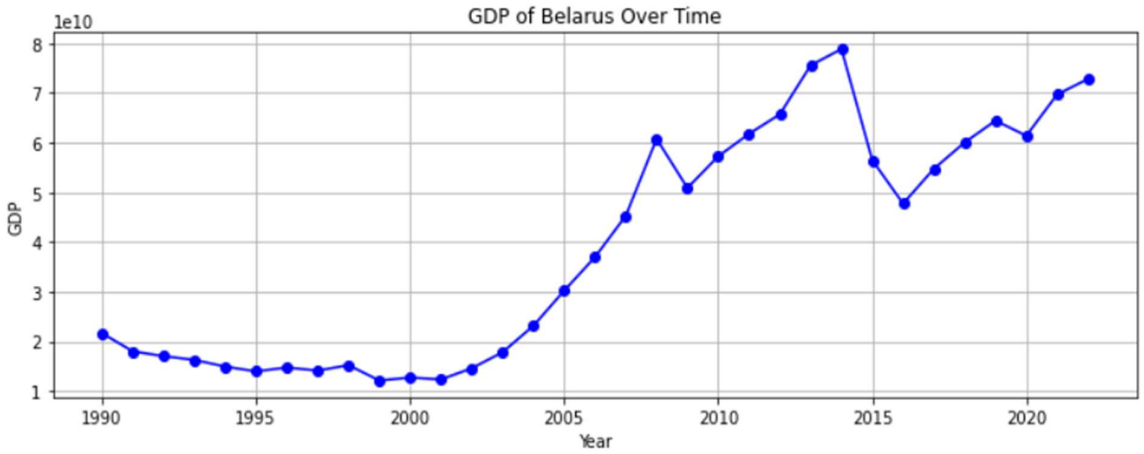


Рисунок 3.1 – ВВП РБ

В качестве рассматриваемых способов прогнозирования в рассмотрение взяты ARIMA, случайный лес, а также градиентный бустинг. Средством реализации выбран язык Python. Для реализации вышеописанных метод, а также построения графиков, в программу были импортированы все необходимые библиотеки, представленные на рисунке 3.2:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error
from sklearn.ensemble import RandomForestRegressor
from xgboost import XGBRegressor
from pmdarima import auto_arima
from statsmodels.tsa.arima.model import ARIMA
import warnings
from statsmodels.tsa.deterministic import DeterministicProcess
```

Рисунок 3.2 – Библиотеки Python для прогнозирования

Для обучения моделей выбран период с 1990 по 2018. Результат построения моделей получился следующий:

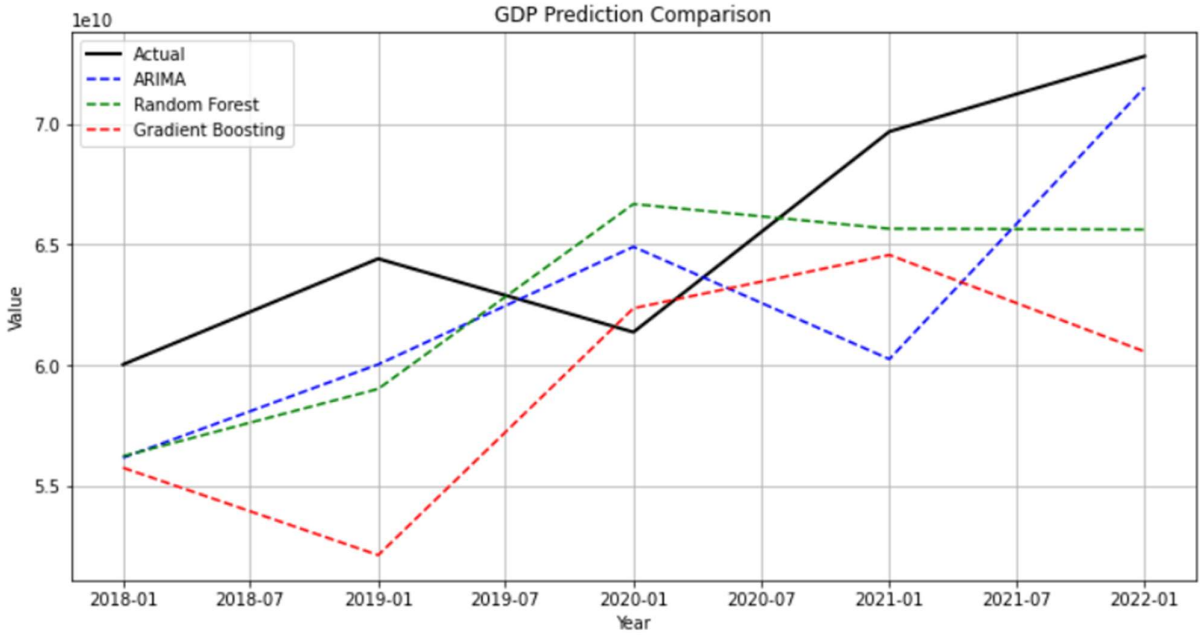


Рисунок 3.3 – Сравнение методов прогнозирования

Для оценки точности предсказаний будем использовать такие показатели, как MSE, MAE и RMSE.

MSE или средне квадратичная ошибка показывает средний квадрат отклонения предсказанного значения от истинного и определяется по формуле (3.1):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 . \quad (3.1)$$

MAE отражает среднее отклонение абсолютного предсказанного значения от истинного и вычисляется по формуле (3.2):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| . \quad (3.2)$$

RMSE же является ничем иным как квадратным корнем из MSE, то есть данный показатель можно вычислить по формуле (3.3):

$$RMSE = \sqrt{MSE}. \quad (3.3)$$

По итогам прогнозирования RMSE и MAE оказались 5 и выше млрд. долларов, что является довольно сильной погрешностью.

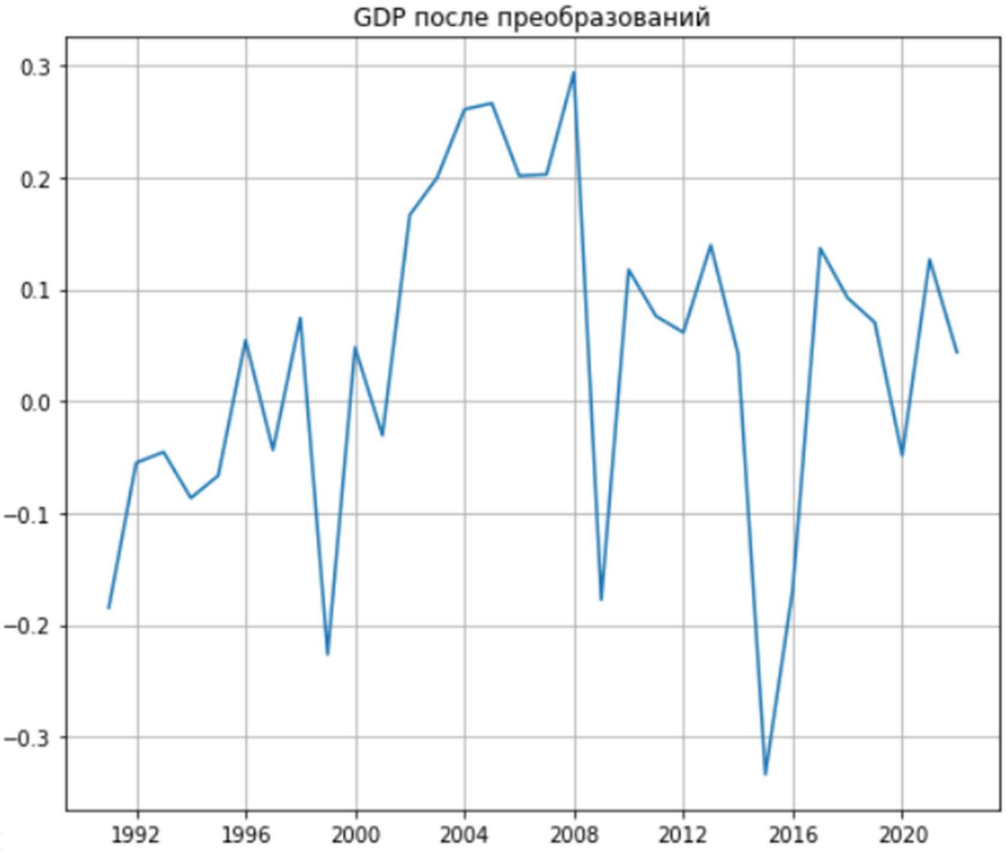
```
GDP Prediction Evaluation:
ARIMA Metrics:
MSE: 27386860280789553152.00
MAE: 4496359375.67
RMSE: 5233245673.65

Random Forest Metrics:
MSE: 27841372127511261184.00
MAE: 5136402038.86
RMSE: 5276492407.60

Gradient Boosting Metrics:
MSE: 69072989909773975552.00
MAE: 6972996817.66
RMSE: 8311016177.93
```

Рисунок 3.4 – Результаты прогноза

Для улучшения точности прогнозов предпримем логарифмирование исходных данных, а также сделаем дифференцирование. Полученный ряд данных ВВП выглядит следующим образом:



я

Рисунок 3.5 – Логарифмированные данные по ВВП

А построенный прогноз вышел следующим:

Рисунок 3.6 – Новый прогноз ВВП

Из рисунка 3.6 видно, что изменения позволили значительно улучшить точность метода ARIMA, об этом и свидетельствует улучшение показателей. Однако в двух методах машинного обучения получилось ухудшение показателей. Сами результаты представлены на рисунке 3.7

```
GDP Prediction Evaluation:
ARIMA Metrics:
MSE: 10284575112888369152.00
MAE: 3176533649.46
RMSE: 3206957298.26

Random Forest Metrics:
MSE: 46516542904337743872.00
MAE: 5586690944.03
RMSE: 6820303725.23

Gradient Boosting Metrics:
MSE: 119209473718301835264.00
MAE: 10026909223.66
RMSE: 10918309105.27
```

Рисунок 3.7 – Результаты нового прогноза

Логарифмирование и дифференцирование хорошо сказались на методе ARIMA, улучшив его точность. Данный метод оказался самым эффективным и его использование возможно на практике. Что касательно методов машинного обучения, то им изменение пошли только в минус. Тут для улучшения точности следует отказаться от логарифмирования, добавить больше признаков, а также увеличить объемы исходной выборки.

Полный код реализации представлен в приложении А.

3.2 Анализ настроений на рынке акций

Как уже упоминалось в работе, определенные ожидания потребителей могут влиять на экономическую ситуацию. Для разбора данного факта проанализируем связь стоимости акций на бирже с ожиданиями людей. В качестве источника настроений будем использовать твиты на экономические темы для соответствующей компании.

Рассмотрим акции компании Apple. В качестве периода анализа возьмем данные с января 2020 по январь 2022 года. Рассматривать будем на ежедневной основе по цене закрытия торгов. За аналогичный период выгрузим данные по

твитам из социальной сети X(Twitter) с хэштегами #AAPL, #Apple, #Share, #Stock и т.п.

После анализа настроений все твиты разделяются на три группы: Bullish (позитивный/позиция быка на рынке), Bearish (негативный/позиция медведя на рынке) и Neutral (нейтральная позиция). И за каждый день определяется наиболее частое настроение.

Результат сопоставления графика цен и настроений представлен на рисунке 3.8

Рисунок 3.8 – Анализ настроений для акций Apple

В качестве проверки полученных результатов используется коэффициент корреляции Пирсона и Спирмена. Первый из них определяется по формуле (3.4):

$$r_{xy} = \frac{\sum(x_i - M_x)(y_i - M_y)}{(n-1)\sigma_x\sigma_y}, \tag{3.4}$$

где x, y – значение исследуемых признаков, M_x, M_y – соответствующие средние значения, σ_x, σ_y – соответствующие стандартные отклонения, n – количество наблюдений. Корреляция Спирмена же определяется по формуле (3.5):

$$r_s = 1 - \frac{6\sum d^2}{n(n^2-1)}, \tag{3.5}$$

где $\sum d^2$ – сумма квадратов разностей наблюдаемых значений.

Результаты получились следующие: корреляция Пирсона 0,283, корреляция Спирмена 0,343.

Также рассмотрим связь настроения и цены с лагом, то есть рассмотрим влияние не день в день, а с задержкой в 1 и в 2 дня. Результат для лага в 1 день следующий: -0.023 (Пирсона) и -0.042(Спирмена). Для лага в два дня – 0.022 (Пирсона) и -0.028 (Спирмена).

Как итог, видно, что настроения имеют краткосрочный эффект влияния на цены и практически не имеют никакого влияния на цены в последующие дни.

Полный код реализации представлен в приложении Б.

ЗАКЛЮЧЕНИЕ

В работе рассмотрены основные макроэкономические показатели, характеризующие состояние экономики. Изучена связь между данными показателями, продемонстрированы способы их расчета. Рассмотрено общее понятие временного ряда и изучены все его компоненты, а также показаны способы их выявления.

Среди методов прогнозирования выделены методы авторегрессии и скользящего среднего ARMA и модификация ARIMA, приведены популярные примеры таких моделей с известными коэффициентами, такими как Марковский процесс, например. Также продемонстрирована роль машинного обучения в современной науке о данных, рассмотрены такие классические методы как случайный лес и градиентный бустинг. Все эти методы были использованы для прогнозирования ВВП Республике Беларусь. Более точным оказался метод ARIMA из-за наличия у ряда данных заметного тренда.

Особое внимание было уделено анализу настроений. Были продемонстрированы принципы работы искусственного интеллекта с текстовыми данными, проанализированы твиты людей и исследовано влияние их настроений на стоимость. Краткосрочный эффект настроение пользователей имеет, однако не более чем в пределах исследуемых суток.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Агапова, Т.А. Макроэкономика: учебник / Т.А. Агапова; под общ. ред. д.э.н., проф. А.В. Сидоровича; МГУ им. М.В. Ломоносова. – 6-е изд., стереотип. – М.: Дело и Сервис, 2004. – 448 с.
2. Алексеева, В.А. Анализ временных рядов: учебное пособие / В.А. Алексеева. – Ульяновск: УлГТУ, 2020. – 147 с.
3. Афанасьев, В.Н. Анализ временных рядов и прогнозирование: учебник / В.Н. Афанасьев, М.М. Юзбашев. – М.: Финансы и статистика, 2001. – 228 с.
4. Богатырев, В.Д. Микроэкономика: учебное пособие / В.Д. Богатырев, А.Ю. Ситникова. – Самара: Изд-во СГАУ, 2015. – 116 с.
5. Бокс, Дж. Анализ временных рядов: прогноз и управление / Дж. Бокс, Г. Дженкинс. – М.: Мир, 1974. – 380 с.
6. Есипова, О.В. Макроэкономическое планирование и прогнозирование: учебное пособие / О.В. Есипова. – Самара: Изд-во Самарского университета, 2018. – 72 с.
7. Мэнкью, Н.Г. Принципы Экономикс / Н.Г. Мэнкью. – СПб.: Питер Ком, 1999. – 784 с.
8. Мухамедиев, Р.И. Введение в машинное обучение: учебник / Р.И. Мухамедиев, Е.Н. Амиргалиев. – Алматы, 2022. – 252 с.
9. Рыбакова, И.В. Макроэкономика: учебное пособие для студентов направления подготовки «Экономика» / И.В. Рыбакова; Рубцовский индустриальный институт. – Рубцовск, 2017. – 127 с.
10. Сакович, М.В. Занятость и безработица // Экономика. Университетский курс: учебное пособие / под ред. П.С. Лемещенко, С.В. Лукина. – Мн.: Книжный Дом, 2007. – С. 400–422.
11. Хасти, Т. Основы машинного обучения / Т. Хасти, Р. Тибширани, Дж. Фридман. – М.: ДМК Пресс, 2020. – 540 с.

ПРИЛОЖЕНИЕ А

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.preprocessing import StandardScaler
from statsmodels.tsa.stattools import adfuller

data = pd.read_csv('D:/Загрузки/GPD By Country.csv')
belarus_data = data[data['country'] == 'Belarus'].copy()
belarus_data['year'] = pd.to_datetime(belarus_data['year'], format='%Y')
belarus_data.set_index('year', inplace=True)
belarus_data.sort_index(inplace=True)

def prepare_series(series):
    log_series = np.log(series)
    diff_series = log_series.diff().dropna()
    adf_result = adfuller(diff_series)
    print(f'ADF Statistic: {adf_result[0]}')
    print(f'p-value: {adf_result[1]}')
    print('Стационарный ряд' if adf_result[1] < 0.05 else 'Ряд не стационарный')

    return log_series, diff_series

belarus_data['GDP_log'], gdp_diff = prepare_series(belarus_data['GDP'])
belarus_data['GDP_pc_log'], gdp_pc_diff = prepare_series(belarus_data['GPD per Capita'])

plt.figure(figsize=(14, 6))
plt.subplot(1, 2, 1)
plt.plot(gdp_diff, label='GDP (лог + дифф)')
plt.title('GDP после преобразований')
plt.grid(True)

plt.subplot(1, 2, 2)
plt.plot(gdp_pc_diff, label='GDP per Capita (лог + дифф)')
plt.title('GDP per Capita после преобразований')
plt.grid(True)
plt.tight_layout()
plt.show()

def create_features(diff_series, original_log_series, lags=3):
    df = pd.DataFrame({'diff': diff_series})
    for lag in range(1, lags+1):
        df[f'lag_{lag}'] = df['diff'].shift(lag)

    df['rolling_mean'] = df['diff'].rolling(window=3).mean()
    df['rolling_std'] = df['diff'].rolling(window=3).std()

    df['log_value'] = original_log_series[df.index]

    return df.dropna()
gdp_features = create_features(gdp_diff, belarus_data['GDP_log'])
X_gdp = gdp_features.drop('diff', axis=1)
y_gdp = gdp_features['diff']

gdp_pc_features = create_features(gdp_pc_diff, belarus_data['GDP_pc_log'])
X_gdp_pc = gdp_pc_features.drop('diff', axis=1)
y_gdp_pc = gdp_pc_features['diff']
test_size = 5
X_train_gdp, X_test_gdp = X_gdp[:-test_size], X_gdp[-test_size:]
y_train_gdp, y_test_gdp = y_gdp[:-test_size], y_gdp[-test_size:]
```

```
63.
64. X_train_gdp_pc, X_test_gdp_pc = X_gdp_pc[:-test_size], X_gdp_pc[-test_size:]
65. y_train_gdp_pc, y_test_gdp_pc = y_gdp_pc[:-test_size], y_gdp_pc[-test_size:]
66.
67. def inverse_transform(log_series, diff_predictions, last_original_value):
68.     log_preds = []
69.     current = last_original_value
70.     for diff in diff_predictions:
71.         current = current + diff
72.         log_preds.append(current)
73.
74.     return np.exp(log_preds)
75.
76. def arima_model(train, test, order=(1,0,1)):
77.     history = list(train)
78.     predictions = []
79.
80.     for t in range(len(test)):
81.         model = ARIMA(history, order=order)
82.         model_fit = model.fit()
83.         yhat = model_fit.forecast()[0]
84.         predictions.append(yhat)
85.         history.append(test.iloc[t])
86.
87.     return predictions
88.
89. arima_gdp_pred_diff = arima_model(y_train_gdp, y_test_gdp, order=(1,0,1))
90. arima_gdp_pred = inverse_transform(belarus_data['GDP_log'], arima_gdp_pred_diff,
91.                                    belarus_data['GDP_log'].iloc[-test_size-1])
92. arima_gdp_pc_pred_diff = arima_model(y_train_gdp_pc, y_test_gdp_pc, order=(1,0,1))
93. arima_gdp_pc_pred = inverse_transform(belarus_data['GDP_pc_log'], arima_gdp_pc_pred_diff,
94.                                       belarus_data['GDP_pc_log'].iloc[-test_size-1])
95. def train_ml_model(model, X_train, y_train, X_test, original_log_series, last_original_val
    ue):
96.     model.fit(X_train, y_train)
97.     diff_pred = model.predict(X_test)
98.     return inverse_transform(original_log_series, diff_pred, last_original_value)
99.
100.        rf_gdp = RandomForestRegressor(n_estimators=100, random_state=42)
101.        gb_gdp = GradientBoostingRegressor(n_estimators=100, random_state=42)
102.
103.        rf_gdp_pred = train_ml_model(rf_gdp, X_train_gdp, y_train_gdp, X_test_gdp,
104.                                     belarus_data['GDP_log'], belarus_data['GDP_log'].iloc[
    -test_size-1])
105.
106.        gb_gdp_pred = train_ml_model(gb_gdp, X_train_gdp, y_train_gdp, X_test_gdp,
107.                                     belarus_data['GDP_log'], belarus_data['GDP_log'].iloc[-
    test_size-1])
108.        rf_gdp_pc = RandomForestRegressor(n_estimators=100, random_state=42)
109.        gb_gdp_pc = GradientBoostingRegressor(n_estimators=100, random_state=42)
110.
111.        rf_gdp_pc_pred = train_ml_model(rf_gdp_pc, X_train_gdp_pc, y_train_gdp_pc, X_test_
    gdp_pc,
112.                                        belarus_data['GDP_pc_log'], belarus_data['GDP_pc_lo
    g'].iloc[-test_size-1])
113.
114.        gb_gdp_pc_pred = train_ml_model(gb_gdp_pc, X_train_gdp_pc, y_train_gdp_pc, X_test_
    gdp_pc,
115.                                        belarus_data['GDP_pc_log'], belarus_data['GDP_pc_log
    '].iloc[-test_size-1])
116.
117.        def evaluate(y_true, y_pred, model_name):
118.            mse = mean_squared_error(y_true, y_pred)
119.            mae = mean_absolute_error(y_true, y_pred)
120.            rmse = np.sqrt(mse)
121.
122.            print(f'{model_name} Metrics:')
```

```
            print(f'MSE: {mse:.2f}')
            print(f'MAE: {mae:.2f}')
            print(f'RMSE: {rmse:.2f}\n')
            return {'MSE': mse, 'MAE': mae, 'RMSE': rmse}
        y_test_gdp_original = belarus_data['GDP'][-test_size:]
        y_test_gdp_pc_original = belarus_data['GPD per Capita'][-test_size:]

        print("GDP Prediction Evaluation:")
        arima_metrics = evaluate(y_test_gdp_original, arima_gdp_pred, 'ARIMA')
        rf_metrics = evaluate(y_test_gdp_original, rf_gdp_pred, 'Random Forest')
        gb_metrics = evaluate(y_test_gdp_original, gb_gdp_pred, 'Gradient Boosting')

        def plot_results(y_true, arima_pred, rf_pred, gb_pred, title):
            plt.figure(figsize=(12, 6))
            plt.plot(y_true.index, y_true, 'ko-', label='Фактические значения')
            plt.plot(y_true.index, arima_pred, 'bo--', label='ARIMA')
            plt.plot(y_true.index, rf_pred, 'go--', label='Random Forest')
            plt.plot(y_true.index, gb_pred, 'ro--', label='Gradient Boosting')
            plt.title(title)
            plt.xlabel('Год')
            plt.ylabel('Значение')
            plt.legend()
            plt.grid(True)
            plt.show()

        plot_results(y_test_gdp_original, arima_gdp_pred, rf_gdp_pred, gb_gdp_pred,
                     'Сравнение методов прогнозирования GDP')
```

ПРИЛОЖЕНИЕ Б

```
1.  import pandas as pd
2.  import numpy as np
3.  import matplotlib.pyplot as plt
4.  import seaborn as sns
5.  from scipy import stats
6.  import statsmodels.api as sm
7.  from statsmodels.tsa.stattools import grangercausalitytests
8.  from statsmodels.stats.outliers_influence import variance_inflation_factor
9.
10. plt.style.use('seaborn-v0_8')
11. def load_and_preprocess_data():
12.     try:
13.         tweets = pd.read_csv("D:/Загрузки/AAPL_pre_processed.csv",
14.                              parse_dates=['created_at'],
15.                              dtype={'body': 'string'})
16.
17.         stock = pd.read_csv("D:/Загрузки/Apple_stock_history.csv",
18.                             parse_dates=['Date'])
19.
20.         sentiment_categories = ['Bullish', 'Bearish', 'Neutral']
21.         tweets['sentiment'] = pd.Categorical(
22.             tweets['sentiment'].fillna('Neutral'),
23.             categories=sentiment_categories
24.         )
25.
26.         sentiment_map = {'Bullish': 1, 'Bearish': -1, 'Neutral': 0}
27.         tweets['sentiment_num'] = tweets['sentiment'].map(sentiment_map).astype('float16')

28.
29.         daily = tweets.resample('D', on='created_at').agg(
30.             sentiment_mean=('sentiment_num', 'mean'),
31.             tweet_count=('body', 'count')
32.         ).reset_index()
33.
34.         merged = pd.merge(daily, stock, left_on='created_at', right_on='Date', how='inner'
    )
35.         merged['daily_return'] = merged['Close'].pct_change()
36.
37.         for lag in [1, 2, 3]:
38.             merged[f'sentiment_lag{lag}'] = merged['sentiment_mean'].shift(lag)
39.             merged[f'return_lag{lag}'] = merged['daily_return'].shift(lag)
40.
41.         merged = merged.replace([np.inf, -np.inf], np.nan).dropna()
42.         return merged
43.
44.     except Exception as e:
45.         print(f"Ошибка при загрузке данных: {e}")
46.         return None
47.
48. def perform_statistical_analysis(df):
49.     results = {}
50.     def safe_correlation(x, y):
51.         with np.errstate(invalid='ignore'):
52.             mask = ~np.isnan(x) & ~np.isnan(y)
53.             if sum(mask) < 3:
54.                 return np.nan, np.nan
55.             return stats.pearsonr(x[mask], y[mask])
56.     for col in ['sentiment_mean', 'sentiment_lag1', 'sentiment_lag2']:
57.         x = df[col]
58.         y = df['daily_return']
59.
60.         pearson, p_pearson = safe_correlation(x, y)
61.         spearman, p_spearman = stats.spearmanr(x, y, nan_policy='omit')
62.
63.         X = sm.add_constant(x)
```

```
        robust_model = sm.RLM(y, X, M=sm.robust.norms.HuberT()).fit()

        results[col] = {
            'pearson': {'corr': pearson, 'p': p_pearson},
            'spearman': {'rho': spearman, 'p': p_spearman},
            'regression': {
                'coef': robust_model.params[1],
                'p_value': robust_model.pvalues[1],
                'ci': robust_model.conf_int().iloc[1].values.tolist()
            }
        }

    granger_results = grangercausalitytests(
        df[['daily_return', 'sentiment_mean']],
        maxlag=3,
        verbose=False
    )

    vif_data = pd.DataFrame()
    vif_data["feature"] = ['sentiment_mean', 'sentiment_lag1', 'sentiment_lag2']
    vif_data["VIF"] = [
        variance_inflation_factor(
            df[['sentiment_mean', 'sentiment_lag1', 'sentiment_lag2']].values, i
        ) for i in range(3)
    ]

    return results, granger_results, vif_data

def visualize_results(df, analysis_results):
    # Установка современной цветовой палитры
    sns.set_palette("husl")
    current_palette = sns.color_palette()

    fig, axes = plt.subplots(3, 1, figsize=(14, 18))

    axes[0].plot(df['Date'], df['Close'],
                         label='Цена AAPL',
                         color=current_palette[0])
            axes[0].set_ylabel('Цена ($)', color=current_palette[0])
            ax2 = axes[0].twinx()
            ax2.plot(df['Date'], df['sentiment_mean'],
                    label='Sentiment',
                    color=current_palette[2],
                    linestyle='--')
            ax2.set_ylabel('Sentiment', color=current_palette[2])
            axes[0].set_title('Динамика цены и настроений в твитах', pad=20)

            sns.scatterplot(
                x='sentiment_mean',
                y='daily_return',
                data=df,
                ax=axes[1],
                color=current_palette[3],
                alpha=0.6
            )
            sns.regplot(
                x='sentiment_mean',
                y='daily_return',
                data=df,
                ax=axes[1],
                scatter=False,
                color=current_palette[4]
            )
            axes[1].set_title('Зависимость дневной доходности от sentiment', pad=15)


            lags_data = pd.DataFrame({
                'Lag': ['Текущий', 'Лаг 1 день', 'Лаг 2 дня'],
```

```
132.                 'Корреляция': [
133.                     analysis_results[0]['sentiment_mean']['pearson']['corr'],
134.                     analysis_results[0]['sentiment_lag1']['pearson']['corr'],
135.                     analysis_results[0]['sentiment_lag2']['pearson']['corr']
136.                 ],
137.                 'P-value': [
138.                     analysis_results[0]['sentiment_mean']['pearson']['p'],
139.                     analysis_results[0]['sentiment_lag1']['pearson']['p'],
140.                     analysis_results[0]['sentiment_lag2']['pearson']['p']
141.                 ]
142.             })
143.
144.             sns.barplot(
145.                 x='Lag',
146.                 y='Корреляция',
147.                 data=lags_data,
148.                 ax=axes[2],
149.                 palette='viridis'
150.             )
151.             axes[2].set_title('Корреляция между sentiment и доходностью по лагам', pad=15)
152.
153.             plt.tight_layout()
154.             plt.show()
155.
156.         def main():
157.             data = load_and_preprocess_data()
158.             if data is None:
159.                 return
160.
161.             stats_results, granger, vif = perform_statistical_analysis(data)
162.
163.             print("\n=== Результаты корреляционного анализа ===")
164.             for name, res in stats_results.items():
165.                 print(f"\n{name.replace('_', ' ')}:")
166.                 print(f"Корреляция Пирсона: {res['pearson']['corr']:.3f} (p={res['pearson'
    ]['p']:.3f})")
167.                 print(f"Корреляция Спирмена: {res['spearman']['rho']:.3f} (p={res['spearma
    n']['p']:.3f})")
168.             for lag in [1, 2, 3]:
169.                 p_val = granger[lag][0]['ssr_ftest'][1]
170.                 print(f"Лаг {lag} день: p-value = {p_val:.4f}")
171.
172.             visualize_results(data, (stats_results, granger, vif))
173.
174.         if __name__ == "__main__":
175.             main()
```