**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**Кафедра компьютерных технологий и систем**

**ОЦЕНИВАНИЕ ПАРАМЕТРОВ МОДЕЛИ ARIMA ПРИ ПРОГНОЗИРОВАНИИ ВРЕМЕННЫХ РЯДОВ**

Курсовой проект

Ёды Никиты Дмитриевича

студента 3 курса, 6 группы

Специальность “прикладная математика”

Научный руководитель:

Старший преподаватель

Шолтанюк Станислав Витальевич

Минск, 2023 г.

## ОГЛАВЛЕНИЕ

1. Основные понятия модели ARIMA . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
2. Предварительный анализ временного ряда . . . . . . . . . . . . . . . . . . . . . . . . . . .5
   1. Временной ряд . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .5
   2. Методы прогнозирования временных рядов . . . . . . . . . . . . . . . . 6
   3. Анализ временного ряда . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .7
3. Параметры модели ARIMA. Методология оценивания значений параметров для конкретного временного ряда . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
4. Проверка модели ARIMA на адекватность. Построение прогноза по модели ARIMA . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .11
5. Сравнительный анализ моделей ARIMA с различными значениями параметров . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .16
6. Заключение. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .19
7. Приложение . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .20
8. Список использованной литературы . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .21

# Основные понятия модели ARIMA

Модель ARIMA (AutoregRessive Integrated Moving Average) – один из наиболее распространённых методов анализа и прогнозирования временных рядов. Эта модель позволяет обработать данные временного ряда, чтобы лучше понять этот ряд или предсказать его развитие

ARIMA использует три основных параметра (p, d, q), которые выражаются целыми числами. Потому модель также записывается как ARIMA(p, d, q). Вместе эти три параметра учитывают сезонность, тенденцию и шум в наборах данных:

* p – порядок авторегрессии (AR), который позволяет добавить предыдущие значения временного ряда.

(1.1)

где – коэффициенты авторегрессии, – случайная ошибка.

* d – порядок интегрирования (I; т. е. порядок разностей исходного временного ряда). Он добавляет в модель понятия разности временных рядов (определяет количество прошлых временных точек, которые нужно вычесть из текущего значения).
* q – порядок скользящего среднего (MA), который позволяет установить погрешность модели как линейную комбинацию наблюдавшихся ранее значений ошибок.

(1.2)

где — параметры модели, – случайная ошибка.

Модель ARIMA имеет следующий вид:

(1.3)

где — оператор разности d— го порядка (— — разности 1— го порядка).

# Предварительный анализ временного ряда

## 2.1. Временной ряд

Временной ряд представляет собой последовательность значений, упорядоченных во времени. Он может быть получен из различных источников, например, финансовых данных, климатических измерений, экономических показателей и т.д. Временные ряды обладают несколькими особыми свойствами, которые влияют на выбор и применение методов прогнозирования:

* Тренд: Тренд представляет собой долгосрочное направление изменения временного ряда. Он может быть восходящим (увеличение значений), нисходящим (уменьшение значений) или плоским (отсутствие явного направления изменения). Тренд может быть линейным или нелинейным.
* Сезонность: Сезонность отражает периодические колебания во временном ряде, которые повторяются в определенные моменты времени, например, ежегодно, ежемесячно или еженедельно. Сезонность может быть добавлена в виде циклических колебаний с фиксированной продолжительностью.
* Шум: Шум представляет случайную компоненту временного ряда, которая не подчиняется определенным закономерностям. Он может возникать из-за различных факторов, таких как случайные воздействия, ошибки измерения и прочее. Шум может вносить непредсказуемость и сложность в анализ временного ряда.

Временной ряд может быть классифицирован как стационарный или нестационарный. Различие между ними заключается в свойствах и характеристиках, которые они обладают:

* Стационарный временной ряд: Стационарный временной ряд обладает постоянными статистическими свойствами во времени. Это означает, что его среднее значение, дисперсия и автокорреляционная функция не меняются со временем. В стационарном ряду отсутствуют тренды и сезонные колебания. Это облегчает анализ и прогнозирование, так как можно предположить, что будущие значения будут подчиняться тем же статистическим свойствам, что и прошлые значения.
* Нестационарный временной ряд: Нестационарный временной ряд, в отличие от стационарного, имеет изменяющиеся статистические свойства во времени. Здесь могут присутствовать тренды, сезонные колебания и другие изменения, которые делают его статистически непостоянным. Нестационарные ряды могут быть более сложными для анализа и прогнозирования, так как их статистические свойства могут меняться со временем.
* При прогнозировании временных рядов предпочтительным является работа с стационарными рядами, так как они обеспечивают более надежные статистические свойства и предсказуемость. В случае нестационарных рядов может потребоваться применение методов преобразования данных или моделей, способных учитывать их нестационарность, чтобы получить более точные прогнозы.

## 2.2. Методы прогнозирования временных рядов

Методы прогнозирования временных рядов являются важным инструментом для анализа и предсказания последовательных данных. Вот некоторые из наиболее распространенных методов прогнозирования временных рядов:

* Метод скользящего среднего (Moving Average, MA): Этот метод основан на вычислении среднего значения временного ряда в заданном окне (обычно фиксированной длины). Он полезен для сглаживания временных рядов и выявления общих тенденций.
* Метод экспоненциального сглаживания (Exponential Smoothing): Этот метод учитывает веса предыдущих наблюдений во временном ряду, применяя экспоненциально убывающие коэффициенты сглаживания. Он позволяет учитывать как тренды, так и сезонность в данных.
* Авторегрессионная интегрированная скользящая средняя (Autoregressive Integrated Moving Average, ARIMA): Это классический и широко используемый метод прогнозирования временных рядов. Он объединяет три компоненты: авторегрессию (AR), интеграцию (I) и скользящее среднее (MA). Модель ARIMA может учитывать тренды, сезонность и шум в данных.
* Методы машинного обучения: В последние годы методы машинного обучения, такие как рекуррентные нейронные сети (RNN) и долгая краткосрочная память (LSTM), стали популярными для прогнозирования временных рядов. Они могут автоматически извлекать сложные зависимости из данных и учитывать долгосрочные зависимости.

## 2.3. Анализ временного ряда

Для работы используется набор данных «Atmospheric CO2 from Continuous Air Samples at Mauna Loa Observatory, Hawaii, U.S.A.», в котором находятся данные о CO2 с марта 1958 года по декабрь 2001 года.

Необходимо предварительно обработать данные. Работа с данными за неделю представляет определенные сложности из-за их короткого временного промежутка, поэтому лучше использовать средние данные за месяц. Для заполнения пропущенных значений во временных рядах можно воспользоваться функцией fillna().

Заполнение средним значением:  
 При использовании данной стратегии пропущенные значения заменяются средним значением всех известных значений в ряде. Для этого вычисляется среднее значение с помощью метода mean(), после чего оно применяется в fillna(). Таким образом, все пропущенные значения в колонке будут заменены на среднее значение.

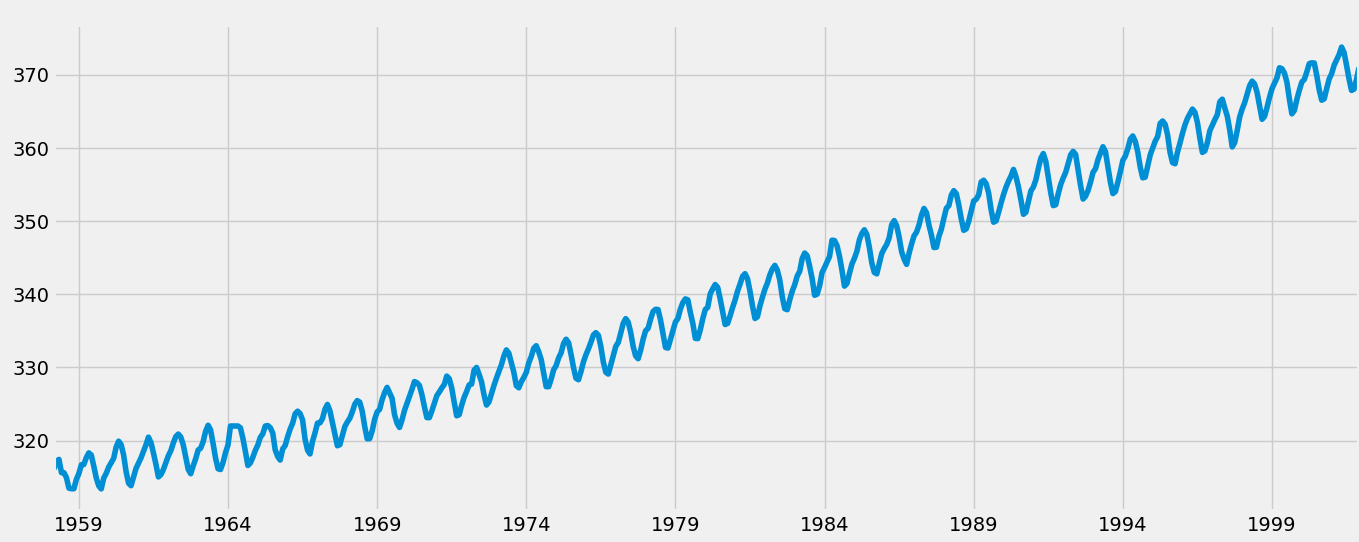


Рисунок 1. - Исходные данные атмосферного содержания CO2

При построении графика на основе этих данных можно заметить некоторые явные шаблоны. Временные ряды имеют очевидную сезонность и общий тренд на увеличение.

# Параметры модели ARIMA. Методология оценивания значений параметров для конкретного временного ряда

Основной задачей при выборе данных для временных рядов в сезонной модели ARIMA является определение оптимальных значений ARIMA (p, d, q) (P, D, Q) s, которые наилучшим образом соответствуют требуемым показателям.

Для итерации по различным комбинациям параметров используется метод сеточного поиска.

Для каждой комбинации параметров функция SARIMAX() из модуля statsmodels может подобрать новую сезонную модель ARIMA и оценить ее общее качество. Оптимальным набором параметров будет тот, в котором модель демонстрирует наиболее высокую производительность по требуемым критериям.

При оценке и сравнении статистических моделей с различными параметрами учитывается, насколько каждая модель соответствует данным и насколько точно она способна прогнозировать будущие значения. В данном случае используется значение AIC (Akaike Information Criterion). AIC оценивает, насколько модель соответствует данным, принимая во внимание её общую сложность. Модель с меньшим количеством параметров, необходимых для достижения соответствия данным, имеет более высокий показатель AIC. Поэтому целью является поиск модели с наименьшим значением AIC.

(3.1)

где k — число параметров модели, L — максимизированное значение функции правдоподобия модели.

Код ниже итерирует комбинации параметров:

warnings.filterwarnings("ignore")  # отключает предупреждения

for param in pdq:

    for param\_seasonal in seasonal\_pdq:

        try:

            mod = sm.tsa.statespace.SARIMAX(y, order=param, seasonal\_order=param\_seasonal, enforce\_stationarity=False, enforce\_invertibility=False)

            results = mod.fit(disp=False)

            print('ARIMA{}x{}12— AIC:{}'.format(param, param\_seasonal, results.aic))

        except:

            Continue

Результат:

ARIMAX(0, 0, 0)x(0, 0, 1, 12) — AIC:6787.3436240402125  
ARIMAX(0, 0, 0)x(0, 1, 1, 12) — AIC:1596.711172764114  
ARIMAX(0, 0, 0)x(1, 0, 0, 12) — AIC:1058.9388921320026  
ARIMAX(0, 0, 0)x(1, 0, 1, 12) — AIC:1056.2878315690562  
ARIMAX(0, 0, 0)x(1, 1, 0, 12) — AIC:1361.6578978064144  
ARIMAX(0, 0, 0)x(1, 1, 1, 12) — AIC:1044.7647912940095  
...  
ARIMAX(1, 1, 1)x(1, 0, 0, 12) — AIC:576.8647112294245  
ARIMAX(1, 1, 1)x(1, 0, 1, 12) — AIC:327.9049123596742  
ARIMAX(1, 1, 1)x(1, 1, 0, 12) — AIC:444.12436865161305  
ARIMAX(1, 1, 1)x(1, 1, 1, 12) — AIC:277.7801413828764

AIC (277.78) — оптимальные параметры.

Необходимо добавить оптимальные параметры в модель ARIMA:

mod = sm.tsa.statespace.SARIMAX(y,  
order=(1, 1, 1),  
seasonal\_order=(1, 1, 1, 12),  
enforce\_stationarity=False,  
enforce\_invertibility=False)  
results = mod.fit()  
print(results.summary().tables[1])

Таблица - значение параметров модели SARMIMA в результате обучения:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | coef | std err | z | P>|z| | [0.025 | 0.975] |
| ar.L1 | 0.3181 | 0.092 | 3.441 | 0.001 | 0.137 | 0.499 |
| ma.L1 | — 0.6254 | 0.077 | — 8.162 | 0.000 | — 0.776 | — 0.475 |
| ar.S.L12 | 0.0010 | 0.001 | 1.732 | 0.083 | — 0.000 | 0.002 |
| ma.S.L12 | — 0.8769 | 0.026 | — 33.806 | 0.000 | — 0.928 | — 0.826 |
| sigma2 | 0.0972 | 0.004 | 22.632 | 0.000 | 0.089 | 0.106 |

ar.L1: Коэффицент авторегрессии (AR)

ma.L1: Коэффицент скользящего среднего (MA)

ar.S.L12: Коэффицент сезонной авторегрессии (SAR)

ma.S.L12: Коэффицент сезонного скользящего среднего (SMA)

sigma2: Оценка дисперсии ошибок модели

coef: Важность каждого параметра и его влияние на временной ряд

str err: Оценка стандартного отклонения коэффициента

z: Мера статистической значимости коэффициента модели

P>|z|: Вероятность получения такого же или более экстремального значения коэффициента

[0.025...0.975]:Промежуток доверительного интервала

Атрибут summary возвращает много информации, но мы сосредоточим наше внимание на таблице коэффициентов. Столбец coef определяет важность каждого параметра и его влияние на временной ряд. Столбец P>|z| сообщает значимость каждого параметра. Здесь вес (важность) каждого параметра p имеет близкое к 0,05 значение, поэтому разумно сохранить в модели все параметры. Диагностика модели и исследование на необычное поведение (Рисунок 2)

# Проверка модели ARIMA на адекватность. Построение прогноза по модели ARIMA

Главная задача – убедиться, что остатки модели некоррелированные и распределяются с нулевым средним значением. Если сезонная модель ARIMA не удовлетворяет этим свойствам, это значит, что ее еще можно улучшить.

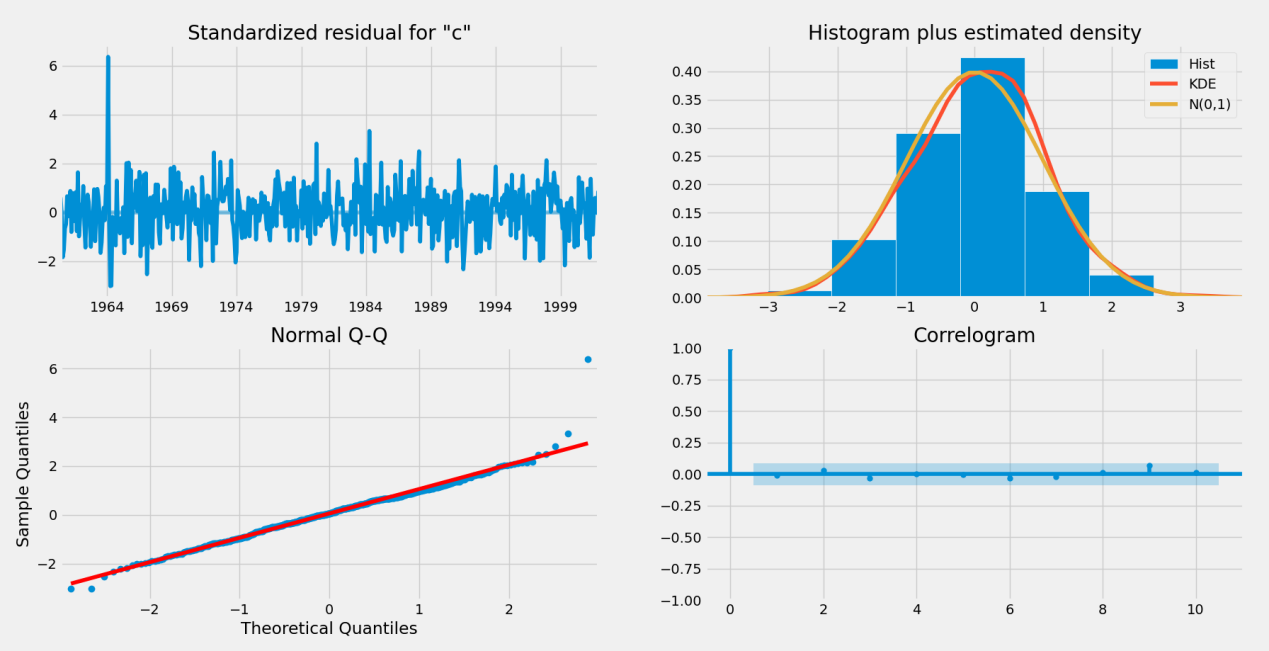


Рисунок 2. - Анализ вычисленных данных

В этом случае диагностика показала, что остатки модели правильно распределяются:

На верхнем правом графике красная линия KDE находится близко к линии N(0,1) (где N(0,1) является стандартным обозначением нормального распределения со средним 0 и стандартным отклонением 1) . Это хороший признак того, что остатки нормально распределены.

[График Q-Q](https://uk.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D1%96%D0%BA_Q-Q)в левом нижнем углу показывает, что упорядоченное распределение остатков (синие точки) следует линейному тренду выборок, взятых из стандартного распределения N(0, 1). Опять же, это признак того, что остатки нормально распределены.

Остатки с течением времени (верхний левый график) не показывают явной сезонности и кажутся белыми шумами. Это подтверждается графиком автокорреляции (внизу справа), который показывает, что остатки временных рядов имеют низкую корреляцию с запаздывающими данными.

Эти графики позволяют сделать вывод о том, что выбранная модель (удовлетворительно) подходит для анализа и прогнозирования данных временных рядов.

Теперь имеем модель временных рядов, с помощью которой можно спрогнозировать данные.

Для начала нужно сравнить прогнозируемые значения с реальными значениями временного ряда, что поможет понять точность прогнозов. Атрибуты get\_prediction() и conf\_int() позволяют получать значения и интервалы для прогнозов временных рядов.

pred = results.get\_prediction(start=pd.to\_datetime('1998— 01— 01'), dynamic=False)  
pred\_ci = pred.conf\_int()

Данный код начнёт прогнозирование с января 1998.

Аргумент dynamic = False включает пошаговое прогнозирование, а это означает, что прогнозы в каждой точке генерируются с использованием полной истории вплоть до этой точки.

Визуализируем реальные и прогнозируемые значения временного ряда CO2 с помощью кода, чтобы оценить, как всё работает:

ax = y['1990':].plot(label='observed')  
pred.predicted\_mean.plot(ax=ax, label='One— step ahead Forecast', alpha=.7)  
ax.fill\_between(pred\_ci.index,  
pred\_ci.iloc[:, 0],  
pred\_ci.iloc[:, 1], color='k', alpha=.2)  
ax.set\_xlabel('Date')  
ax.set\_ylabel('CO2 Levels')  
plt.legend()  
plt.show()

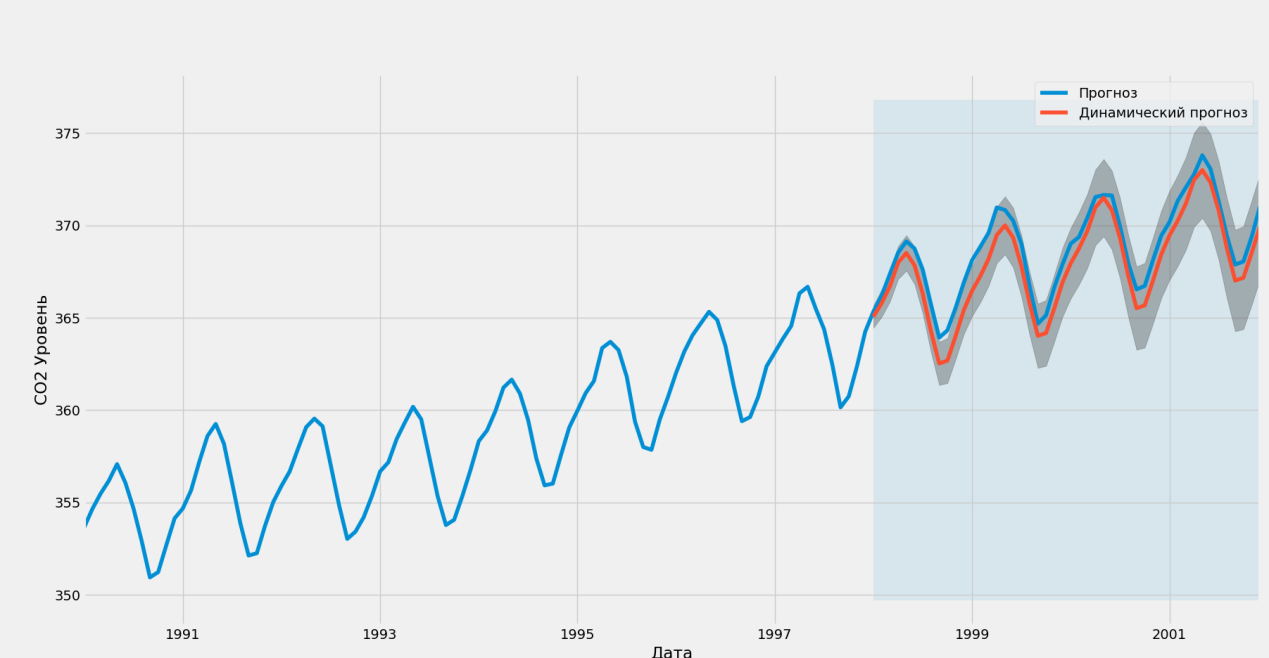


Рисунок 3. - Сравнение фактических и прогнозируемых данных

В целом, прогнозы соответствуют истинным значениям, демонстрируя общий тренд на увеличение.

Также полезно оценить точность наших прогнозов. Для этого можно использовать MSE (Mean Squared Error), что суммирует среднюю ошибку прогнозов. Для каждого прогнозируемого значения нужно вычислить его расстояние до истинного значения. Результаты нужно возводить в квадрат, чтобы различия не компенсировали друг друга при вычислении общего среднего.

y\_forecasted = pred.predicted\_mean  
y\_truth = y['1998— 01— 01':] # Compute the mean square error  
mse = ((y\_forecasted — y\_truth) \*\* 2).mean()  
print('Средняя квадратическая ошибка прогнозов равна:

{}'.format(round(mse, 2)))  
Средняя квадратическая ошибка прогнозов равна: 0.07

MSE прогнозов на один шаг вперед дает значение 0,07 (это очень низкое значение, так как оно близко к 0). MSE 0 означает, что прогноз составлен с идеальной точностью. К этому результату и нужно стремиться, но его не всегда возможно достичь.

Теперь можно использовать модель ARIMA для прогнозирования будущих значений.

Атрибут get\_forecast() объекта временного ряда может вычислить значения на указанное количество шагов.

# Получить прогноз на 500 шагов вперёд  
pred\_uc = results.get\_forecast(steps=500)  
# Получить интервал прогноза  
pred\_ci = pred\_uc.conf\_int()

Этот код можно использовать для визуализации временного ряда и прогнозирования его значений.

ax = y.plot(label='observed', figsize=(20, 15))  
pred\_uc.predicted\_mean.plot(ax=ax, label='Forecast')  
ax.fill\_between(pred\_ci.index,  
pred\_ci.iloc[:, 0],  
pred\_ci.iloc[:, 1], color='k', alpha=.25)  
ax.set\_xlabel('Date')  
ax.set\_ylabel('CO2 Levels')  
plt.legend()  
plt.show()

И сгенерированные прогнозы, и связанный с ними интервал теперь можно использовать для дальнейшего анализа и прогнозирования временных рядов. Полученные данные показывают, что временные ряды будут продолжать стабильный рост.

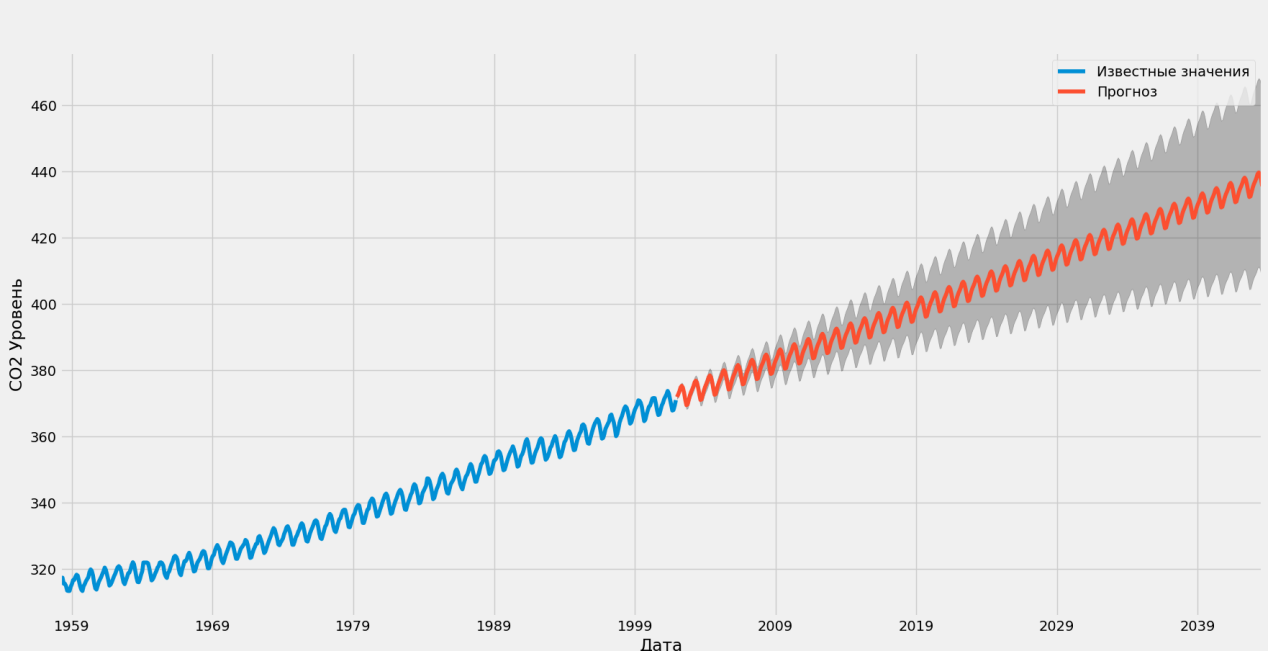


Рисунок 4. - Прогноз изменения параметра

Полученные данные показывают, что временные ряды будут продолжать стабильный рост.

Чем дальше строится прогноз, тем менее точны его значения. Это отражается на интервалах, генерируемых моделью (чем дальше прогноз, тем больше интервал).

# Сравнительный анализ моделей ARIMA с различными значениями параметров

Возьмём параметры: ARIMA(0, 0, 0)x(0, 1, 0, 12)12— AIC:1854.8282341411787

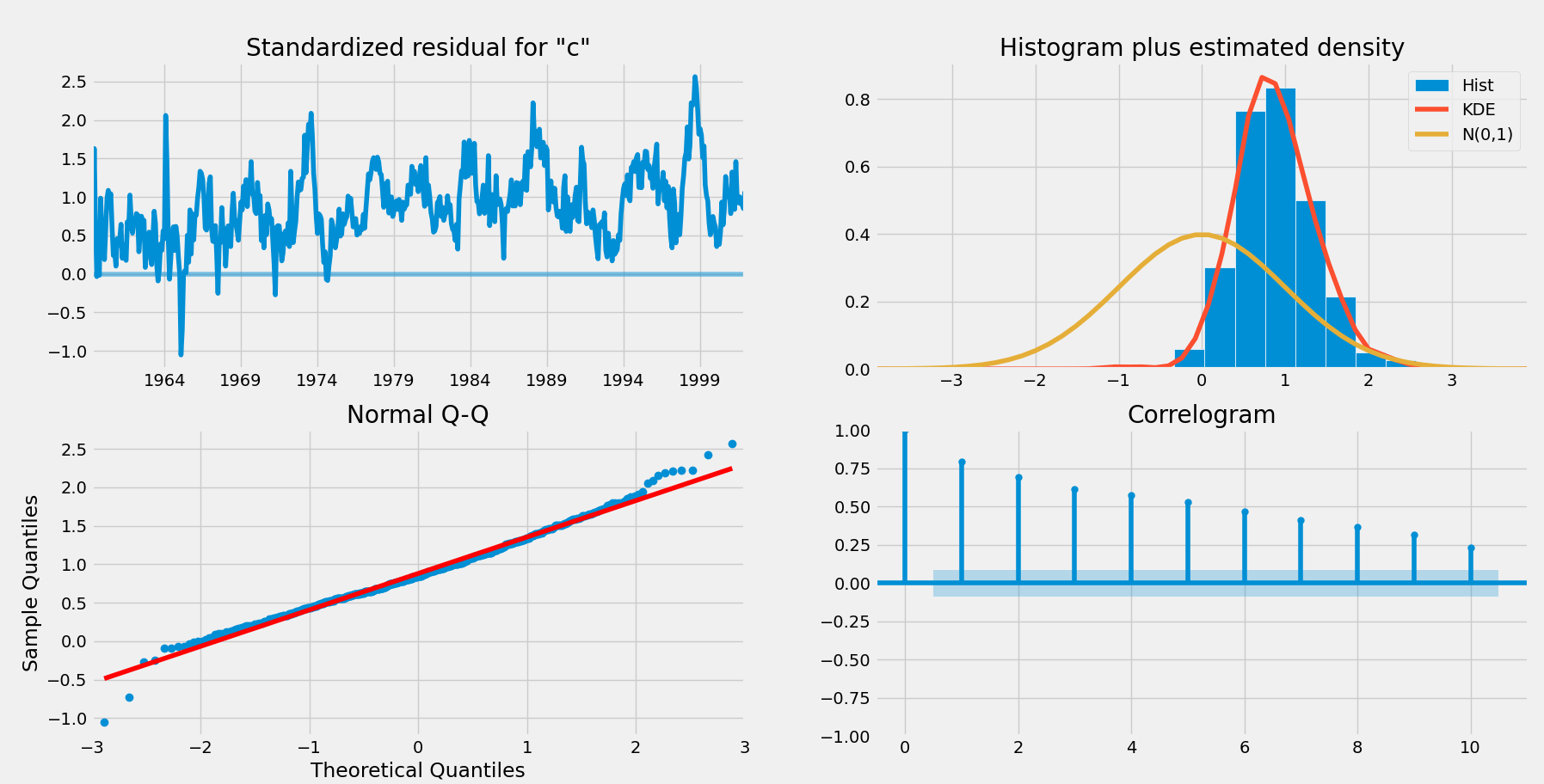


Рисунок 5. - Анализ вычисленных данных

На верхнем правом графике красная линия KDE находится далеко к линии N(0,1) . Это плохой признак того, что остатки нормально распределены.

[График Q-Q](https://uk.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D1%96%D0%BA_Q-Q)в левом нижнем углу показывает, что упорядоченное распределение остатков следует линейному тренду выборок, взятых из стандартного распределения N(0, 1). Это признак того, что остатки нормально распределены.

Остатки с течением времени (верхний левый график) показывают сезонность. Это подтверждается графиком автокорреляции (внизу справа), который показывает, что остатки временных рядов имеют высокую корреляцию.

Эти графики позволяют сделать вывод о том, что выбранная модель (неудовлетворительна) не подходит для анализа и прогнозирования данных временных рядов.

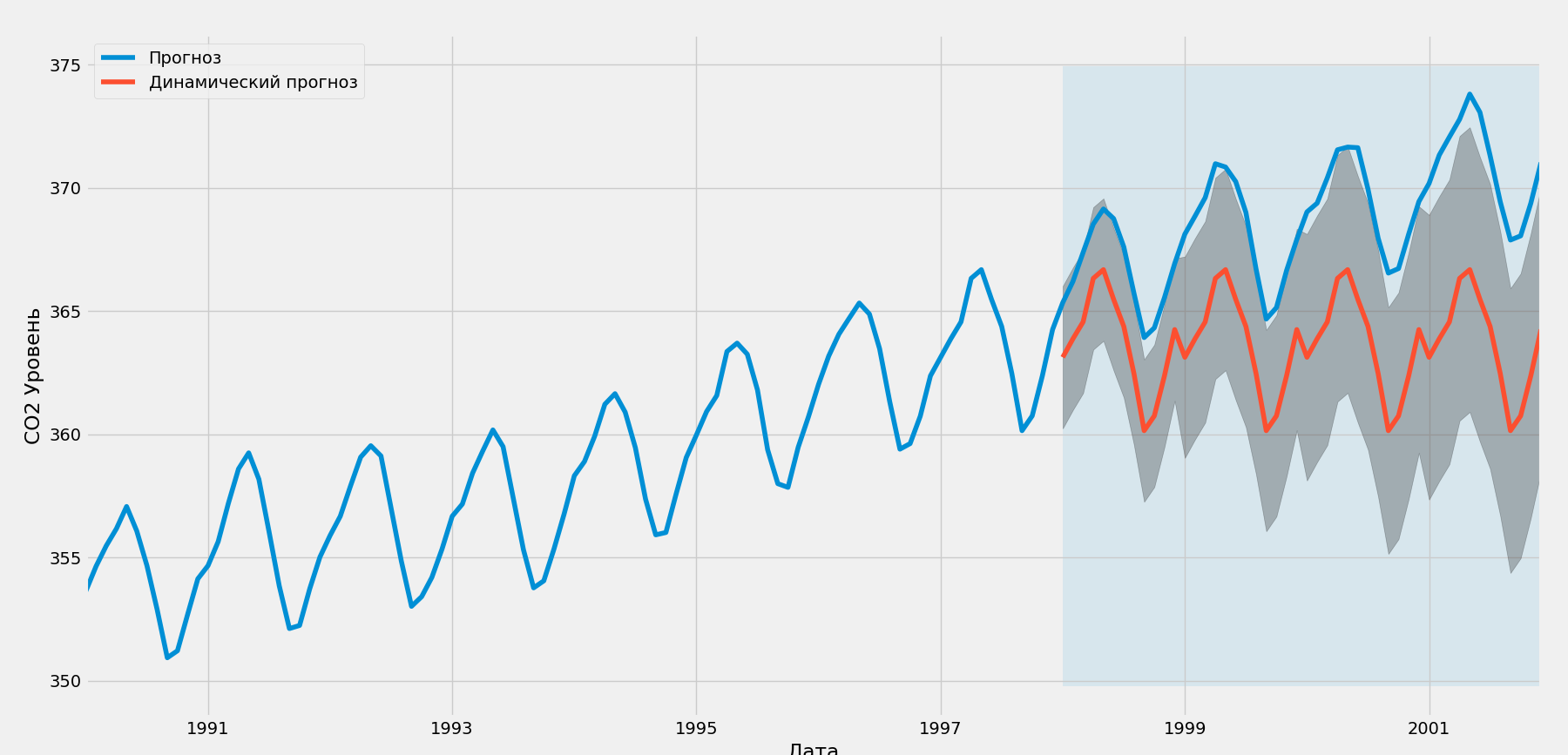


Рисунок 6. - Сравнение фактических и прогнозируемых данных

Прогнозы не соответствуют истинным значениям.

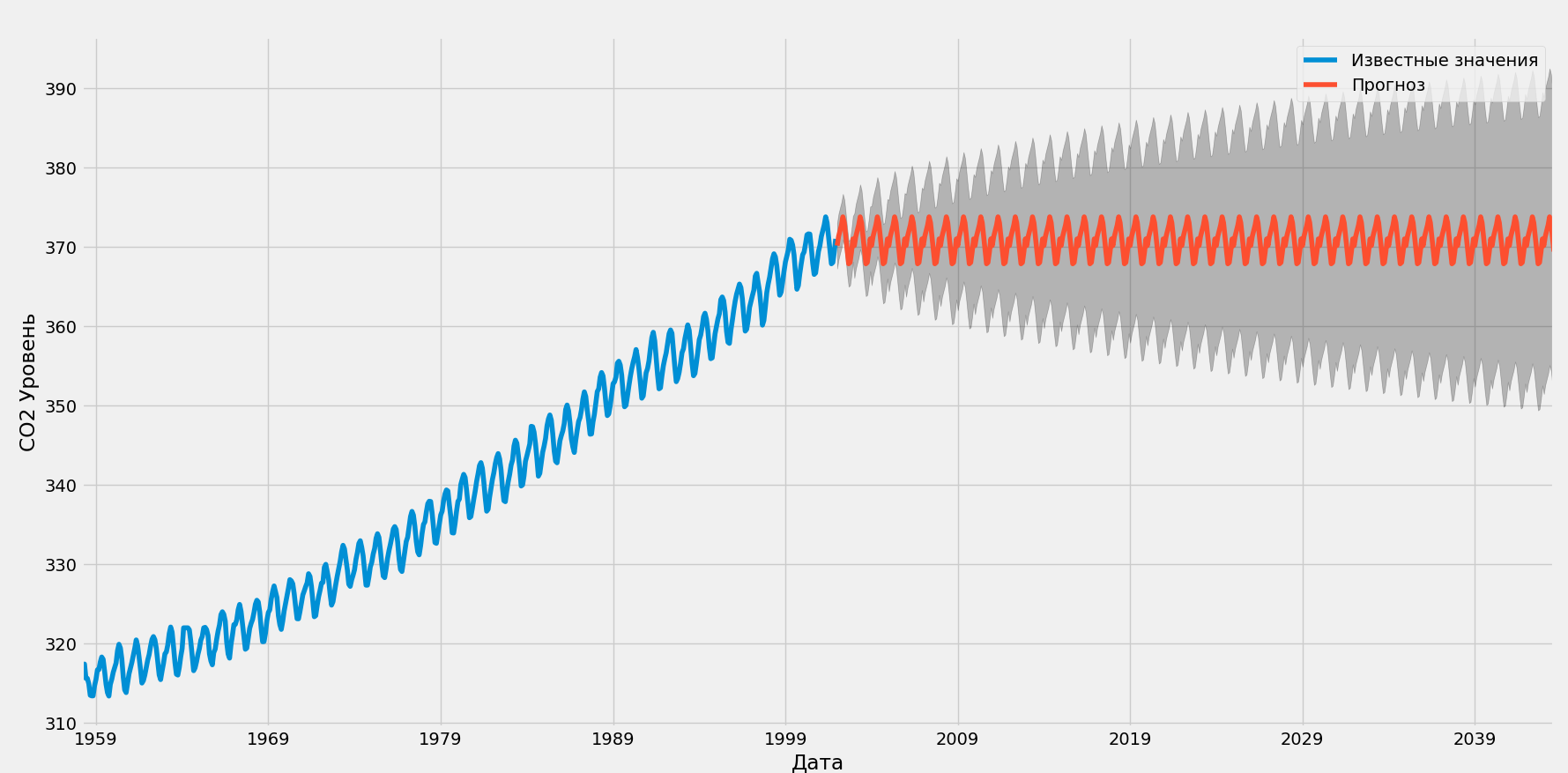


Рисунок 7. - Прогноз изменения параметра

Полученные данные показывают, что временные ряды будут идти по прямой, что говорит нам о том, что параметры для построения графика не верные.

Средняя квадратическая ошибка прогнозов равна: 28.22. Далеко от нуля, значит прогноз неверный.

# ЗАКЛЮЧЕНИЕ

Таким образом, модель ARIMA представляет собой мощный инструмент для прогнозирования временных рядов. Она учитывает авторегрессию, интегрирование и скользящую среднюю, позволяя модели учесть зависимости и шаблоны в данных. ARIMA модель особенно полезна при работе с нестационарными временными рядами, так как она может интегрировать ряды и привести их к стационарному виду.

Однако, стоит отметить, что ARIMA модель предполагает линейные зависимости между значениями ряда и может быть менее эффективной для сложных нелинейных временных рядов. Кроме того, выбор оптимальных значений параметров модели может быть нетривиальной задачей и требовать анализа автокорреляционной функции и частной автокорреляционной функции ряда.

В заключение, модель ARIMA представляет собой важный инструмент для анализа и прогнозирования временных рядов. Ее использование позволяет учесть зависимости в данных и получать достаточно точные прогнозы. Дальнейшее исследование и углубление в методику ARIMA может помочь расширить понимание временных рядов и улучшить качество прогнозирования в различных областях, таких как финансы, экономика, климатология и другие.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Айвазян, С.А. Прикладная статистика: в 3 т. / С.А. Айвазян и др. — М. : Финансы и статистика , 1983–1989. — Т. 2: Исследование зависимостей: Справ. изд. / С.А. Айвазян, И.С. Енюков, Л.Д. Мешалкин; под ред. С.А. Айвазяна. — 487 с.
2. Магнус, Я.Р. Эконометрика. Начальный курс: Учеб. / Я.Р. Магнус, П.К. Катышев, А.А. Пересецкий — 8 — е изд., испр. — М. : Дело, 2007. — 576 с.
3. Hyndman, R.J. Forecasting: principles and practice [Electronic resource] / R.J. Hyndman, G. Athanasopoulos. — 2nd ed. — OTexts: Melbourne, Australia, 2018. — Mode of access: <https://otexts.com/fpp2/>.
4. Бокс, Дж. Анализ временных рядов. Прогноз и управление / Дж. Бокс, Г. Дженкинс: пер. с англ. / Под ред. В.Ф. Писаренко. — М.: Мир, 1974, кн. 1. — 406 с.

**ПРИЛОЖЕНИЕ А**

**ЛИСТИНГ ПРОГРАММЫ**

import warnings

import itertools

import pandas as pd

import numpy as np

import statsmodels.api as sm

import matplotlib.pyplot as plt

plt.style.use('fivethirtyeight')

data = sm.datasets.co2.load\_pandas()

y = data.data

# 'MS' группирует месячные данные

y = y['co2'].resample('MS').mean()

# bfill значит, что нужно использовать значение до заполнения пропущенных значений

y = y.fillna(y.bfill())

print("Данные:")

print(y)

y.plot(figsize=(15, 6))

plt.show()

# Определяю p, d и q в диапазоне 0— 2

print()

p = d = q = range(0, 2)

print("Значения p,d,q:")

print("p: ", p)

print("d: ", d)

print("q: ", q)

# Генерируем различные комбинации p, q и q

print()

print("Все возможные комбинации p,d,q:")

pdq = list(itertools.product(p, d, q))

print("pdq:", pdq)

# Сгенерируйте комбинации сезонных параметров p, q и q

print()

seasonal\_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d, q))]

print('Примеры комбинаций параметров для сезонного ARIMA:')

print('SARIMAX: {} x {}'.format(pdq[1], seasonal\_pdq[1]))

print('SARIMAX: {} x {}'.format(pdq[1], seasonal\_pdq[2]))

print('SARIMAX: {} x {}'.format(pdq[2], seasonal\_pdq[3]))

print('SARIMAX: {} x {}'.format(pdq[2], seasonal\_pdq[4])

print()

print("Вычисление минимального (оптимального) AIC для ARIMA:")

warnings.filterwarnings("ignore")  # отключает предупреждения

for param in pdq:

    for param\_seasonal in seasonal\_pdq:

        try:

            mod = sm.tsa.statespace.SARIMAX(y, order=param, seasonal\_order=param\_seasonal, enforce\_stationarity=False, enforce\_invertibility=False)

            results = mod.fit(disp=False)

            print('ARIMA{}x{}12— AIC:{}'.format(param, param\_seasonal, results.aic))

        except:

            Continue

# Определение модели временных рядов ARIMA

mod = sm.tsa.statespace.SARIMAX(y,

order=(1, 1, 1),

seasonal\_order=(1, 1, 1, 12), enforce\_stationarity=False, enforce\_invertibility=False)

results = mod.fit(disp=False)

print()

print(results.summary().tables[1])

print("ar.L1: \tКоэффицент авторегрессии (AR)")

print("ma.L1: \tКоэффицент скользящего среднего (MA)")

print("ar.S.L12:\tКоэффицент сезонной авторегрессии (SAR)")

print("ma.S.L12:\tКоэффицент сезонного скользящего среднего (SMA)")

print("sigma2: \tОценка дисперсии ошибок модели")

print("coef: \tВажность каждого параметра и его влияние на временной ряд")

print()

print("str err: \tОценка стандартного отклонения коэффициента")

print("z: \tМера статистической значимости коэффициента модели")

print("P>|z|: \tВероятность получения такого же или более экстремального значения коэффициента")

print("[0.025...0.975]:Промежуток доверительного интервала")

results.plot\_diagnostics(figsize=(15, 12))

plt.show()

# Прогнозирование временных рядов

pred\_dynamic = results.get\_prediction(start=pd.to\_datetime('1998— 01— 01'), dynamic=True, full\_results=True)

pred\_dynamic\_ci = pred\_dynamic.conf\_int()

ax = y['1990':].plot(label='Прогноз', figsize=(20, 15))

pred\_dynamic.predicted\_mean.plot(label='Динамический прогноз', ax=ax)

ax.fill\_between(pred\_dynamic\_ci.index, pred\_dynamic\_ci.iloc[:, 0], pred\_dynamic\_ci.iloc[:, 1], color='k', alpha=.25)

ax.fill\_betweenx(ax.get\_ylim(), pd.to\_datetime('1998— 01— 01'), y.index[— 1],

alpha=.1, zorder=— 1)

ax.set\_xlabel('Дата')

ax.set\_ylabel('CO2 Уровень')

plt.legend()

plt.show()

# Извлечь прогнозируемые и истинные значения временного ряда

y\_forecasted = pred\_dynamic.predicted\_mean

y\_truth = y['1998— 01— 01':] # Вычислить среднеквадратичную ошибку

mse = ((y\_forecasted — y\_truth) \*\* 2).mean()

print()

print('Средняя квадратическая ошибка прогнозов равна: {}'.format(round(mse, 2)))

# Создание и визуализация прогноза

# Получить прогноз на 500 шагов вперёд

pred\_uc = results.get\_forecast(steps=500)

# Получить интервал прогноза

pred\_ci = pred\_uc.conf\_int()

ax = y.plot(label='Известные значения', figsize=(20, 15))

pred\_uc.predicted\_mean.plot(ax=ax, label='Прогноз')

ax.fill\_between(pred\_ci.index, pred\_ci.iloc[:, 0], pred\_ci.iloc[:, 1], color='k', alpha=.25)

ax.set\_xlabel('Дата')

ax.set\_ylabel('CO2 Уровень')

plt.legend()

plt.show()