

**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**

Курсовая работа

Ёды Никиты Дмитриевича
студента 3 курса
Научный руководитель:
старший преподаватель
Шолтанюк С.В.

Минск, 2024

ОГЛАВЛЕНИЕ

| | |
|--|-----------|
| ВВЕДЕНИЕ | 3 |
| ГЛАВА 1. МОДЕЛЬ ARIMA | 4 |
| 1.1 Структура модели. Основные понятия | 4 |
| 1.2 Сферы применения модели ARIMA в анализе временных рядов и прогнозировании | 5 |
| ГЛАВА 2. ВРЕМЕННОЙ РЯД | 6 |
| 2.1 Основные понятия на примере конкретного временного ряда | 6 |
| 2.2 Методы прогнозирования | 8 |
| 2.3 Обработка данных временного ряда | 9 |
| ГЛАВА 3. ПАРАМЕТРЫ МОДЕЛИ ARIMA | 10 |
| 3.1 Методология оценивания значений параметров для конкретного временного ряда | 10 |
| 3.2 Проверка модели ARIMA на адекватность | 12 |
| ГЛАВА 4. ПРОГНОЗИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ МОДЕЛИ ARIMA И АНАЛИЗ ПАРАМЕТРОВ | 14 |
| 4.1 Построение прогноза используя модель ARIMA | 14 |
| 4.2 Использование некорректных параметров для модели | 17 |
| ГЛАВА 5. ИСПОЛЬЗОВАНИЕ ДРУГИХ ДАННЫХ ДЛЯ МОДЕЛИ ARIMA | 20 |
| 5.1 Временной ряд BTC. | 20 |
| 5.2 Начальные данные | 21 |
| 5.3 Анализ временного ряда | 23 |
| 5.4 Построение прогноза | 25 |
| ГЛАВА 6. ПОСТРОЕНИЕ ПРОГНОЗА РАЗЛИЧНЫХ ВРЕМЕННЫХ РЯДОВ | 27 |
| 6.1 Временной ряд USD | 27 |
| 6.2 Временной ряд количества пассажиров, перевезенных авиакомпаниями | 29 |
| ЗАКЛЮЧЕНИЕ | 31 |
| СПИСОК ИСТОЧНИКОВ | 32 |
| ПРИЛОЖЕНИЕ | 33 |
| Приложение А: листинг программы (модель ARIMA) | 33 |
| Приложение Б: листинг программы (обработка данных) | 36 |

ВВЕДЕНИЕ

Одним из распространенных методов прогнозирования временных рядов является модель ARIMA (Autoregressive Integrated Moving Average). Эта модель комбинирует авторегрессионные, интегрированные и скользящие средние компоненты, что позволяет учитывать как автокорреляцию, так и тренды в данных.

Цель данной работы заключается в исследовании модели ARIMA и ее применении в анализе временных рядов и прогнозировании. Работа включает в себя обзор структуры модели ARIMA и основных понятий, а также рассмотрение сфер применения этой модели.

Также рассматривается понятие временного ряда и его основные характеристики. Это включает методы прогнозирования временных рядов и обработку данных для анализа.

Особое внимание уделяется параметрам модели ARIMA и методологии их оценивания для конкретных временных рядов. Рассматривается также проверка модели ARIMA на адекватность.

Далее рассмотрены процесс прогнозирования с использованием модели ARIMA и анализ параметров. Происходит построение прогноза и изучение влияния некорректных параметров на результаты модели.

Показано применение модели ARIMA на примере конкретного временного ряда - временного ряда BTC, CO2. Анализируются начальные данные, проводится анализ временного ряда и строится прогноз.

Кроме того, исследуется построение прогноза для других временных рядов, таких как USD и количество пассажиров, перевезенных авиакомпаниями.

В заключении подводятся итоги работы, обобщаются результаты и делаются выводы о применимости модели ARIMA в анализе временных рядов и прогнозировании.

ГЛАВА 1. МОДЕЛЬ ARIMA

1.1 Структура модели. Основные понятия

Модель ARIMA (Autoregressive Integrated Moving Average) – один из наиболее распространённых методов анализа и прогнозирования временных рядов. Модель позволяет обработать данные временного ряда, чтобы лучше понять 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.

$$Y_t = \sum_{i=1}^p \alpha_i Y_{t-i} + \varepsilon_t, \quad (1.1)$$

где $\alpha_1, \alpha_2, \dots, \alpha_p$ – коэффициенты авторегрессии, ε_t – 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.

$$Y_t = \sum_{j=1}^q \beta_j \varepsilon_{t-j}, \quad (1.2)$$

где $\beta_1, \beta_2, \dots, \beta_q$ – параметры модели, ε_t – случайная ошибка.

Модель ARIMA имеет следующий вид:

$$\Delta^d Y_t = \sum_{i=1}^p \alpha_i \Delta^d Y_{t-i} + \sum_{j=1}^q \beta_j \varepsilon_{t-j} + \varepsilon_t, \quad (1.3)$$

где Δ^d – оператор разности d – го порядка ($\Delta Y_t = Y_t - Y_{t-1}$ – разности 1 – го порядка).

1.2 Сферы применения модели ARIMA в анализе временных рядов и прогнозировании

Модель ARIMA широко используется в анализе временных рядов и прогнозировании в различных областях. Вот некоторые из них:

- **Финансовая аналитика:** Модель ARIMA применяется для анализа и прогнозирования финансовых временных рядов, таких как цены акций, валютные курсы, индексы рынков и другие финансовые инструменты.
- **Эконометрика:** ARIMA используется для моделирования экономических временных рядов и прогнозирования экономических показателей, таких как ВВП, инфляция, безработица и другие макроэкономические переменные.
- **Планирование и спросование:** Модель ARIMA может быть применена для прогнозирования спроса на товары и услуги, помогая в планировании производства, управлении запасами и оптимизации поставок.
- **Климатический анализ:** ARIMA используется для анализа и прогнозирования климатических данных, таких как температура, осадки, уровень воды и другие показатели, что помогает в понимании климатических тенденций и разработке моделей для принятия решений в сельском хозяйстве, энергетике и экологии.
- **Маркетинг и продажи:** ARIMA может быть использована для анализа временных рядов в сфере маркетинга и продаж, помогая в прогнозировании продаж, эффективности рекламных кампаний, сезонности спроса и поведения потребителей.
- **Транспорт и логистика:** ARIMA может быть применена для анализа и прогнозирования транспортных и логистических временных рядов, таких как объемы перевозок, пробки, времена доставки и другие параметры, что помогает в планировании и оптимизации транспортных систем.

Это лишь несколько примеров областей, в которых модель ARIMA может быть полезной. Она может быть применена практически в любой ситуации, где есть временные данные и необходимо анализировать и прогнозировать их поведение.

ГЛАВА 2. ВРЕМЕННОЙ РЯД

2.1 Основные понятия на примере конкретного временного ряда

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.

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. Пример временного ряда:

| | |
|------------|------------|
| 1958-03-01 | 316.100000 |
| 1958-04-01 | 317.200000 |
| 1958-05-01 | 317.433333 |
| ... | ... |
| 2001-10-01 | 368.050000 |
| 2001-11-01 | 369.375000 |
| 2001-12-01 | 371.020000 |

Рисунок 1 - Временной ряд CO₂

Построим график из табличных значений:

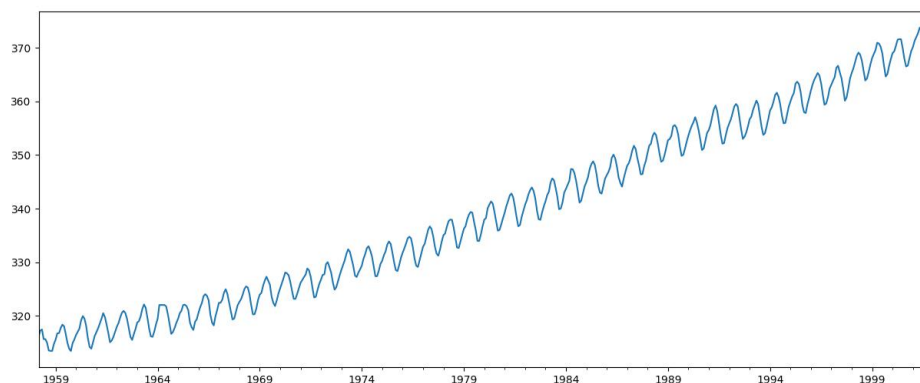


Рисунок 2 - График временного ряда

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.

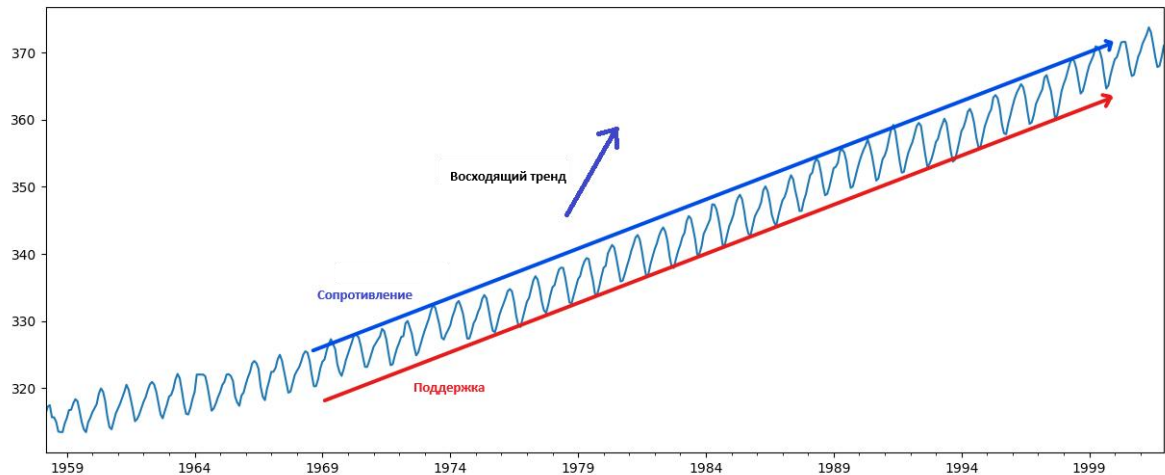


Рисунок 3 - График тренда

- **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.



Рисунок 4 - Сезонная компонента

- **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

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.
- **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 Обработка данных временного ряда

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()`.

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

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.

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

При построении графика на основе этих данных можно заметить некоторые явные шаблоны. Временные ряды имеют очевидную сезонность и общий тренд на увеличение (Рисунок 1)

ГЛАВА 3. ПАРАМЕТРЫ МОДЕЛИ ARIMA

3.1 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.

$$AIC = 2k - 2\ln(L) \quad (3.1)$$

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(dispatch=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])

```

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      Коэффициент авторегрессии (AR)
ma.L1      Коэффициент скользящего среднего (MA)

```

| | |
|-----------------|--|
| ar.S.L12 | Коэффициент сезонной авторегрессии (SAR) |
| ma.S.L12 | Коэффициент 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)

3.2 Проверка модели ARIMA на адекватность

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.

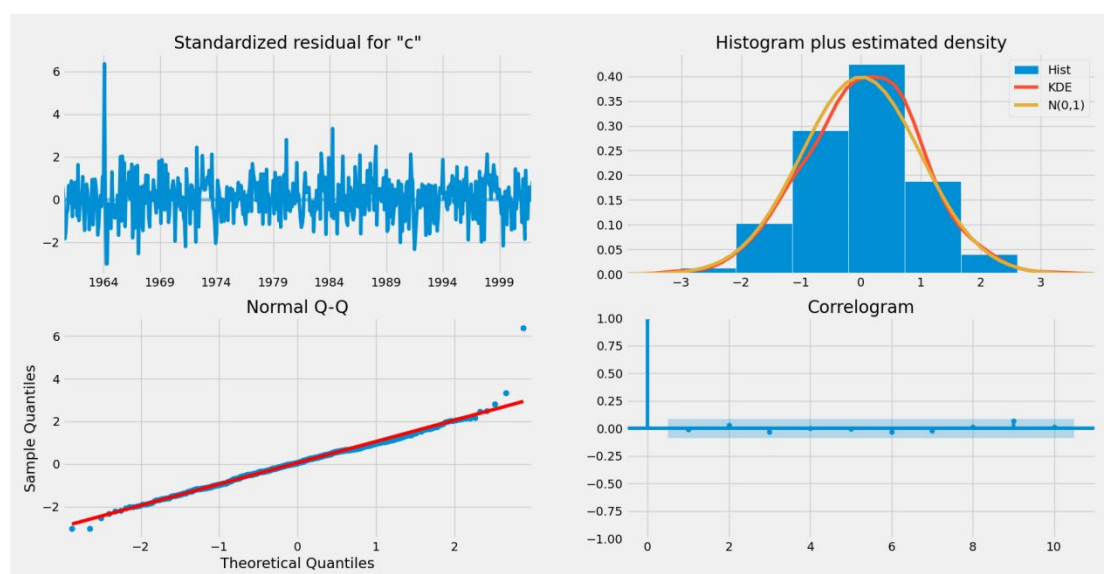


Рисунок 5 - Анализ вычисленных данных

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 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.

ГЛАВА 4. ПОСТРОЕНИЕ ПРОГНОЗА ИСПОЛЬЗУЯ МОДЕЛЬ ARIMA

4.1 Построение прогноза используя модель ARIMA

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 года.

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()
```

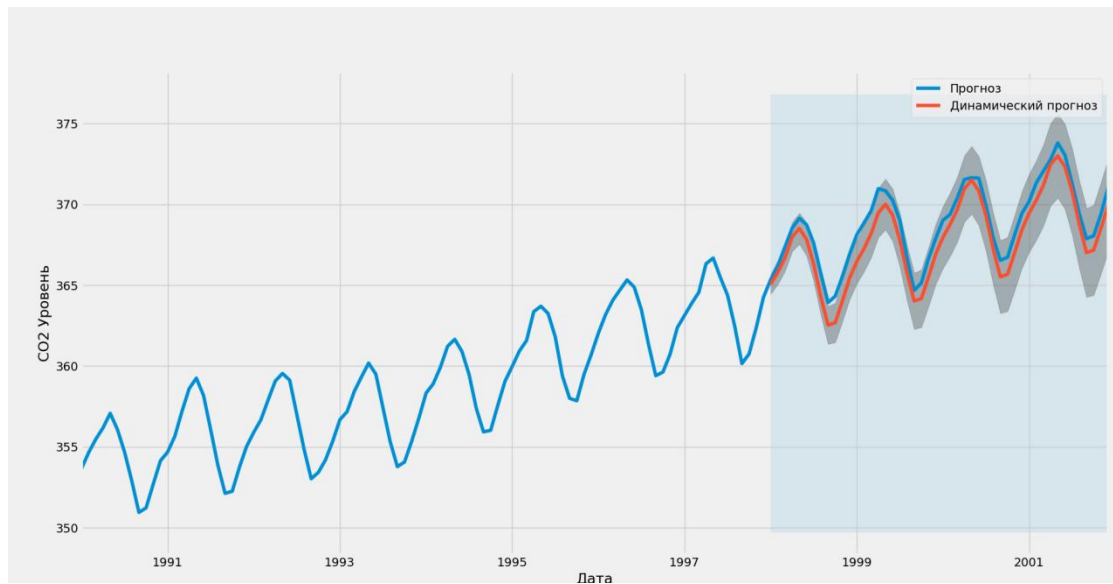


Рисунок 6 - Сравнение фактических и прогнозируемых данных

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.

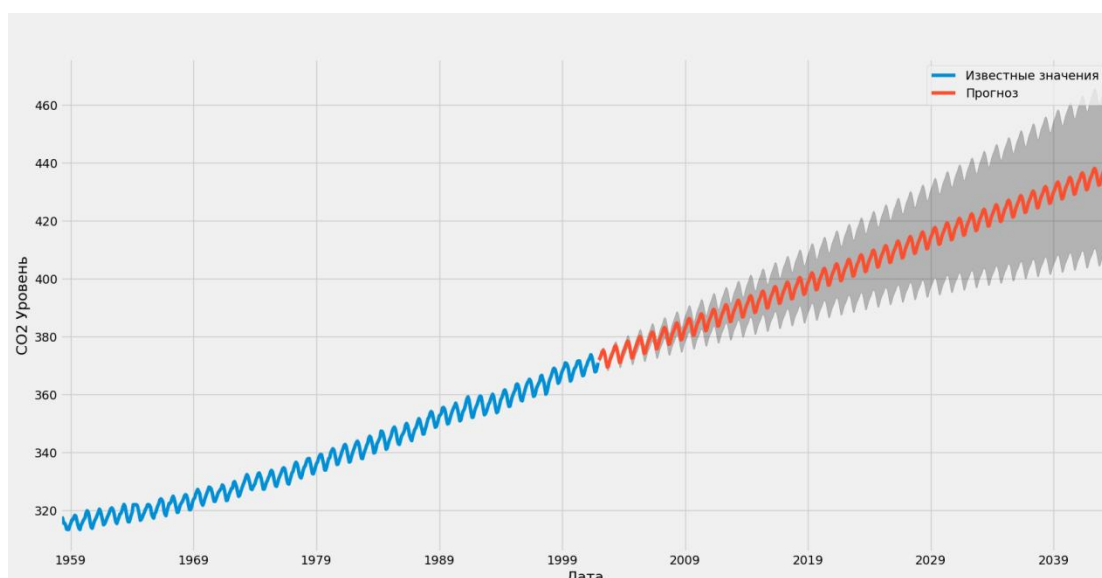


Рисунок 7 - Прогноз изменения параметра

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).

4.2 Использование некорректных параметров для модели ARIMA

Возьмём параметры: $ARIMA(0, 0, 0) \times (0, 1, 0, 12)_{12}$ —
AIC:1854.8282341411787

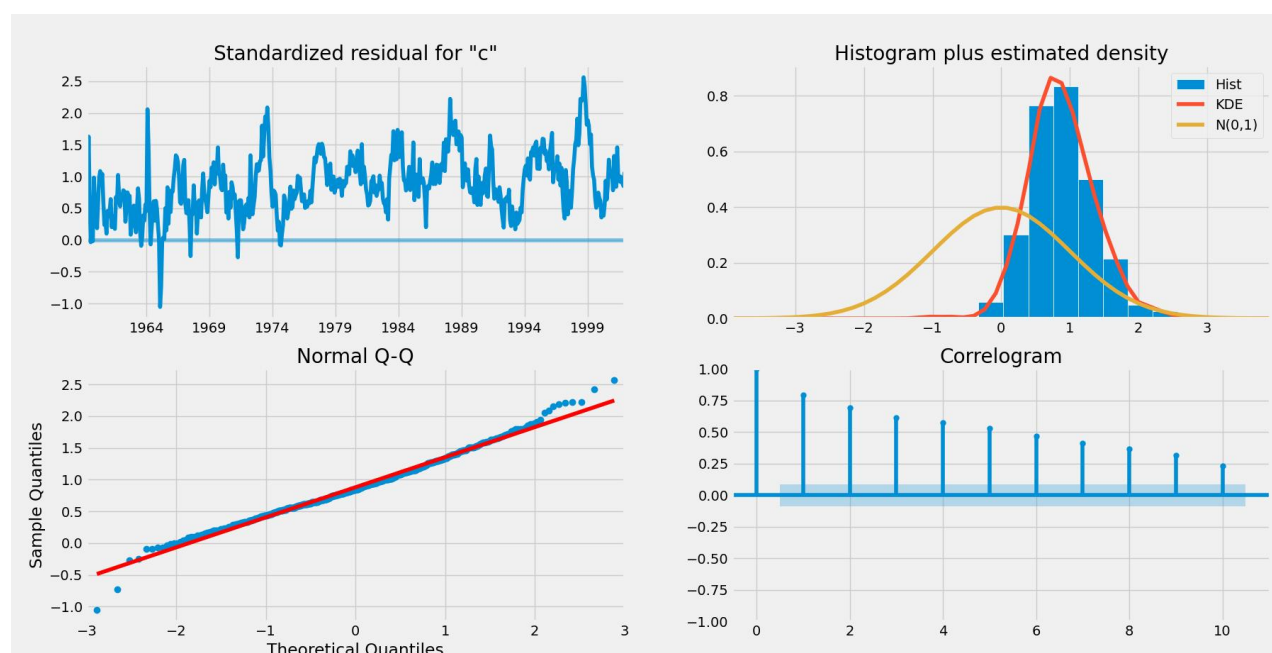


Рисунок 8 - Анализ вычисленных данных

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 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.

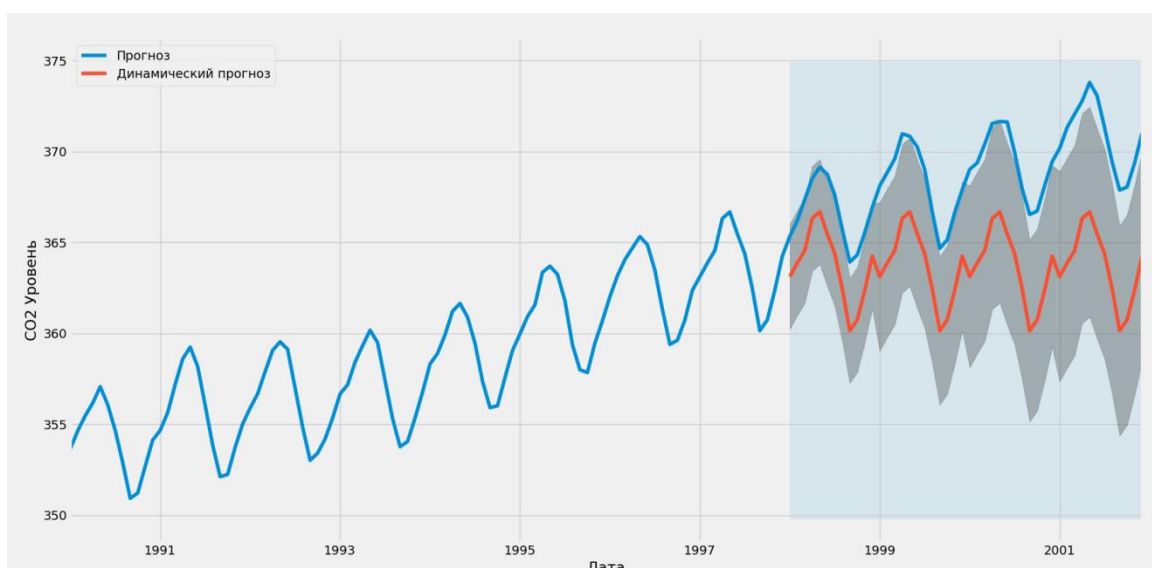


Рисунок 9 - Сравнение фактических и прогнозируемых данных

Прогнозы не соответствуют истинным значениям.

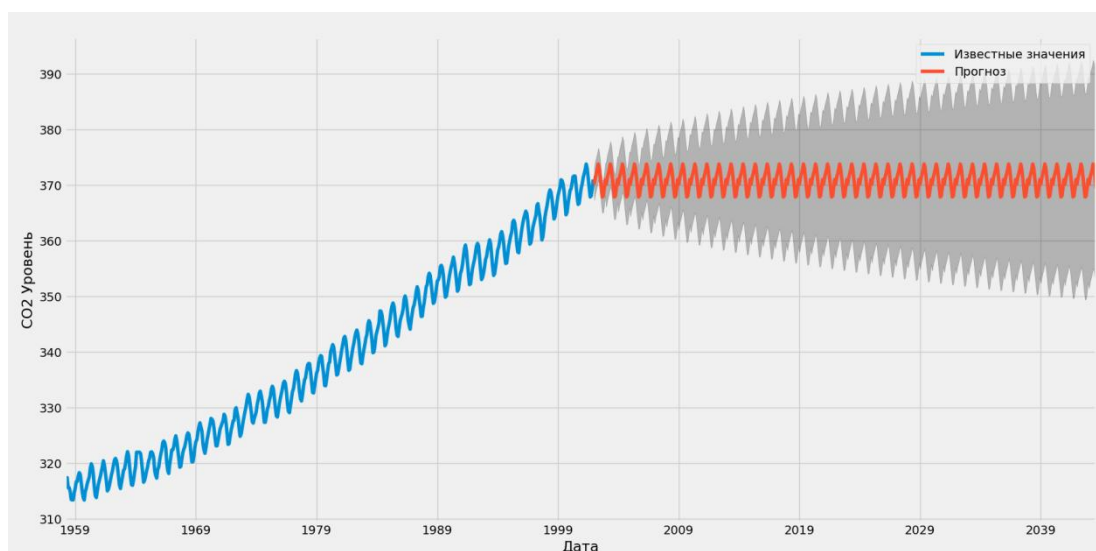


Рисунок 10 - Прогноз изменения параметра

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.

ГЛАВА 5. ИСПОЛЬЗОВАНИЕ ДРУГОГО ВРЕМЕННОГО РЯДА

5.1 Временной ряд BTC

Я использовал временной ряд биткоина для прогнозирования его будущего роста. Временной ряд биткоина представляет собой последовательность изменений его цены на протяжении определенного временного периода. Этот ряд обладает несколькими особенностями, которые я учел при анализе:

Во-первых, я учел высокую волатильность биткоина. Эта криптовалюта известна своей способностью сильно колебаться в цене в короткие промежутки времени. Волатильность создает как возможности для высоких прибылей, так и риски для инвесторов и трейдеров.

Во-вторых, я учел тенденции роста и падения во временном ряду биткоина. Биткоин может проходить через периоды быстрого роста, за которыми следуют снижения цены. Эти тенденции обычно связаны с различными факторами, такими как изменения в регулировании, новости о криптовалюте или инвестиционная активность.

Также, я учел возможную сезонность в цене биткоина. Некоторые исследователи отмечают, что цена биткоина может быть выше в конце года или в начале нового года из-за увеличенного интереса инвесторов.

Кроме того, я учел влияние внешних факторов на цену биткоина. Регулирование криптовалют, новости о хакерских атаках, изменения в политике и экономической ситуации различных стран могут оказывать существенное влияние на цену биткоина и формирование его временного ряда.

Используя эти особенности временного ряда биткоина, я стремлюсь предсказать его будущий рост. Однако, важно отметить, что прогнозирование временных рядов, особенно таких, как биткоин, является сложной задачей, и результаты прогнозов могут быть неточными. Для более точного прогнозирования требуется использование специализированных методов анализа временных рядов и учет различных факторов, влияющих на рынок криптовалют.

5.2 Начальные данные

Временной ряд предоставляет исторические данные о цене биткоина на различных временных интервалах, что позволяет мне изучить его прошлое поведение и выявить тенденции.

Для обработки, перебора и сортировки данных временного ряда биткоина, я разработал специальную программу. Эта программа была написана на языке программирования Java.

Программа позволяет мне загружать и импортировать данные о цене биткоина, предоставленные веб-сайтом.

Входные данные:

| | | | | | |
|------------|---------|---------|----------|---------|---------|
| 01.05.2024 | 91,1418 | 93,0525 | 93,7475 | 90,8618 | -2.05% |
| 01.04.2024 | 93,0525 | 92,4800 | 94,5263 | 91,2425 | +0.62% |
| 01.03.2024 | 92,4800 | 91,1500 | 93,4350 | 90,0970 | +1.46% |
| 01.02.2024 | 91,1500 | 90,0175 | 93,2849 | 89,8360 | +1.26% |
| 01.01.2024 | 90,0175 | 90,3600 | 92,4475 | 87,3827 | -0.38% |
| 01.12.2023 | 90,3600 | 89,4525 | 93,5296 | 88,8773 | +1.01% |
| 01.11.2023 | 89,4525 | 93,4000 | 93,7412 | 87,6858 | -4.23% |
| 01.10.2023 | 93,4000 | 97,9675 | 102,2742 | 91,6394 | -4.66% |
| 01.09.2023 | 97,9675 | 96,0250 | 98,5710 | 92,5561 | +2.02% |
| 01.08.2023 | 96,0250 | 91,6025 | 101,7356 | 91,4490 | +4.83% |
| 01.07.2023 | 91,6025 | 89,5500 | 93,7452 | 88,2854 | +2.29% |
| 01.06.2023 | 89,5500 | 81,1025 | 89,6863 | 80,4990 | +10.42% |
| 01.05.2023 | 81,1025 | 80,2500 | 81,3508 | 75,2510 | +1.06% |
| 01.04.2023 | 80,2500 | 77,5975 | 83,4458 | 77,3763 | +3.42% |
| 01.03.2023 | 77,5975 | 74,9700 | 77,9058 | 74,8825 | +3.50% |
| 01.02.2023 | 74,9700 | 69,8200 | 76,4350 | 69,8076 | +7.38% |
| 01.01.2023 | 69,8200 | 69,9000 | 72,4513 | 67,0998 | -0.11% |

Рисунок 11 - Входные данные

Данные после обработки:

| | |
|------------|-------|
| 2015-08-01 | 64,27 |
| 2015-09-01 | 65,45 |
| 2015-10-01 | 63,94 |
| 2015-11-01 | 66,48 |
| 2015-12-01 | 73,59 |
| ... | ... |
| 2024-01-01 | 90,01 |
| 2024-02-01 | 91,15 |
| 2024-03-01 | 92,48 |
| 2024-04-01 | 93,05 |

Рисунок 12 - Данные после обработки

С помощью программы я построил график, на котором по оси X отображаются месяцы, а по оси Y отображается среднее значение цены биткоина за каждый месяц. График позволяет визуально анализировать динамику цены биткоина и выявлять возможные тенденции и особенности. Это помогает мне понять прошлое поведение цены биткоина и использовать эту информацию для прогнозирования его будущего роста:

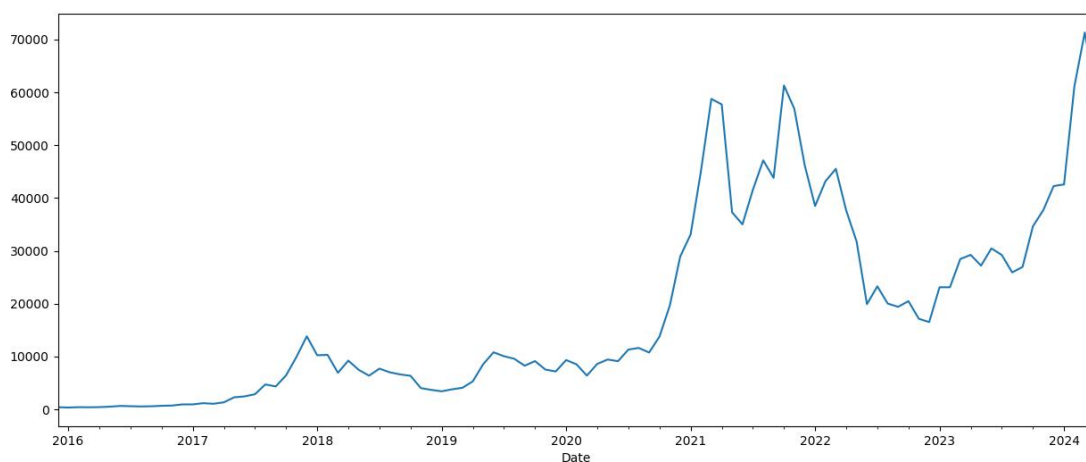


Рисунок 13 - График BTC

5.3 Анализ временного ряда

- **Нормальное распределение:** При проверке нормальности распределения цен биткоина (используя, например, тест Шапиро-Уилка), обычно обнаруживается отклонение от нормальности. Цены биткоина имеют тяжелые хвосты, что означает, что экстремальные значения встречаются чаще, чем в нормальном распределении. Возможны смещение и асимметрия в распределении цен.

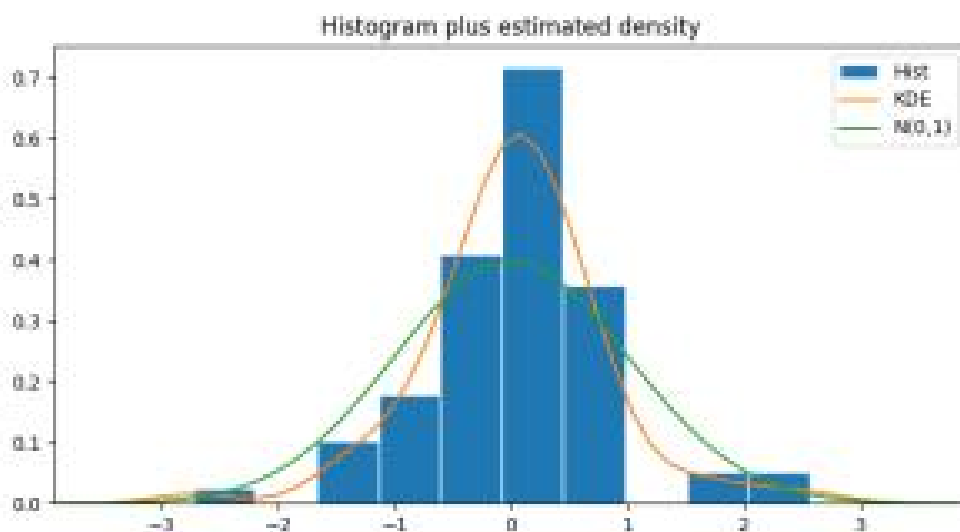


Рисунок 14 - График нормального распределения

- **Тренд:** Анализ тренда в ценах биткоина показывает, что у него присутствует долгосрочная восходящая тенденция. За последние годы биткоин продемонстрировал значительный рост, хотя существует волатильность и коррекции внутри этой общей восходящей трендовой структуры.
- **Шум:** Временной ряд биткоина содержит значительный уровень шума, вызванного краткосрочной волатильностью и эмоциональной реакцией рынка. Этот шум может быть вызван новостями, регулированием, событиями в криптовалютной индустрии и другими внешними факторами. Шум может приводить к краткосрочным колебаниям цены, которые не отражают общую трендовую структуру.

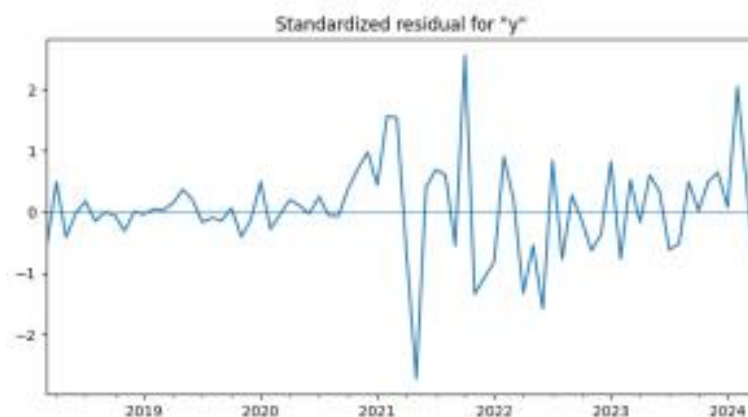


Рисунок 15 - График шума

- Коррелограмма:** Коррелограмма временного ряда биткоина позволяет оценить зависимость между текущим значением цены и предыдущими значениями на разных лагах времени. Возможно обнаружение автокорреляции на некоторых лагах, что указывает на наличие некоторой памяти в ценовых данных. Однако, также может наблюдаться случайность и отсутствие значимой корреляции на некоторых лагах.

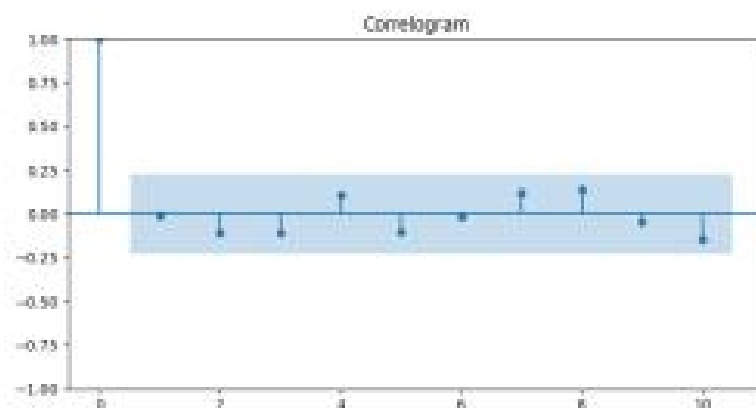


Рисунок 16 - График коррелограммы

- Сезонность:** Сезонность во временном ряду биткоина отсутствует. Сезонность определяет периодические колебания в данных, которые повторяются с постоянным интервалом времени. Однако, в случае биткоина, нет чёткого сезонного паттерна или цикличности, которые можно было бы выделить в анализе временного ряда.

5.4 Построение прогноза

В итоге, с использованием начальных данных, анализа и работы модели ARIMA мы можем предсказывать поведение временного ряда биткоина на основе его исторических значений. Однако, важно учитывать, что прогнозирование финансовых временных рядов, включая цены биткоина, всегда сопряжено с неопределенностью и риском, и требует внимательной интерпретации и дополнительного анализа с учетом других факторов, влияющих на рынок криптовалют.

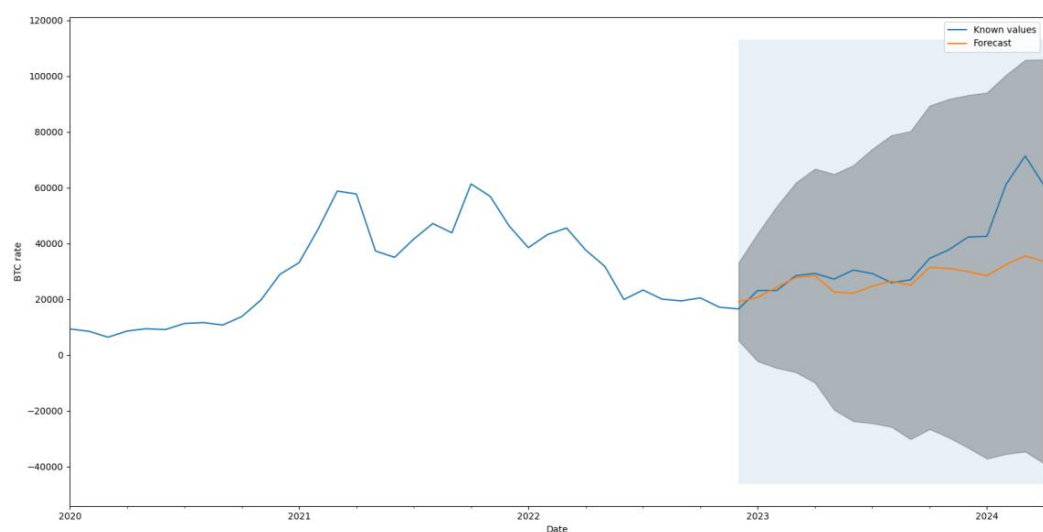


Рисунок 17 - Построение прогноза на участке графика

Построение дальнейшего прогноза:

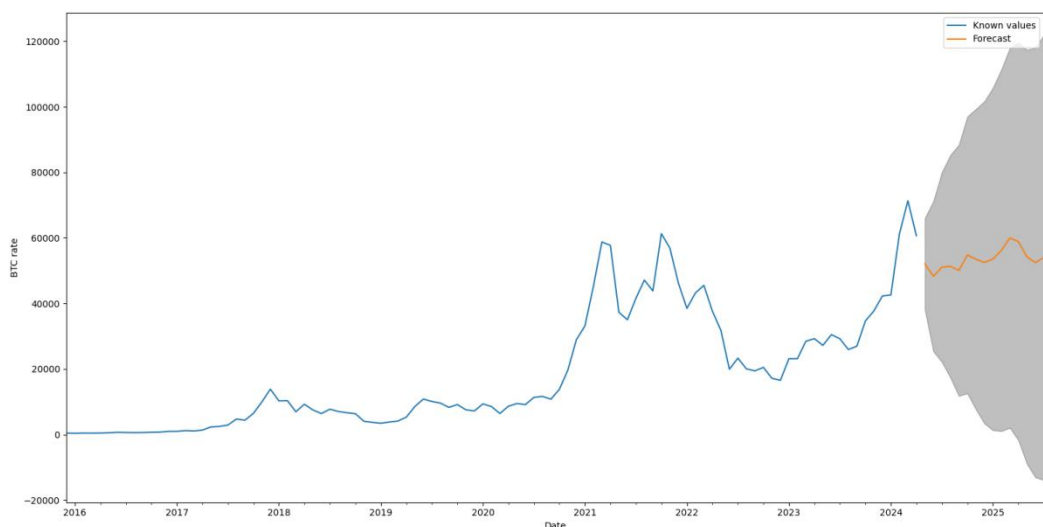


Рисунок 18 - Продолжение графика используя прогноз

Прогнозирование временного ряда биткоина (BTC) является сложной задачей, требующей использования различных методов и подходов. Один из распространенных методов - модель ARIMA (авторегрессионная интегрированная скользящая средняя), позволяет учесть авторегрессионные зависимости и скользящие средние для предсказания поведения ряда.

Однако, важно понимать, что модель ARIMA может иметь свои ограничения, особенно при прогнозировании цен биткоина, который известен своей высокой волатильностью и подвержен влиянию различных факторов, включая новости и события на рынке. Поэтому, для более точного прогнозирования цен биткоина, могут использоваться и другие методы, такие как модели машинного обучения или комбинированные подходы, учитывающие дополнительные факторы и тренды.

Важно отметить, что прогнозирование финансовых временных рядов всегда сопряжено с неопределенностью и риском. Поэтому, полученные прогнозы следует интерпретировать с осторожностью и дополнительно анализировать ситуацию на рынке, учитывая другие факторы, которые могут влиять на цены биткоина.

В целом, прогнозирование временного ряда биткоина является сложной задачей, и для достижения наиболее точных результатов рекомендуется использовать несколько методов и учитывать широкий спектр факторов, включая начальные данные, анализ и моделирование с использованием ARIMA, а также другие методы прогнозирования и актуальную информацию о рынке.

ГЛАВА 6. ПОСТРОЕНИЕ ПРОГНОЗА ДРУГИХ ВРЕМЕННЫХ РЯДОВ

6.1 Временной ряд USD

Входные данные:

| | | | | | |
|------------|---------|---------|----------|---------|---------|
| 01.05.2024 | 91,1418 | 93,0525 | 93,7475 | 90,8618 | -2.05% |
| 01.04.2024 | 93,0525 | 92,4800 | 94,5263 | 91,2425 | +0.62% |
| 01.03.2024 | 92,4800 | 91,1500 | 93,4350 | 90,0970 | +1.46% |
| 01.02.2024 | 91,1500 | 90,0175 | 93,2849 | 89,8360 | +1.26% |
| 01.01.2024 | 90,0175 | 90,3600 | 92,4475 | 87,3827 | -0.38% |
| 01.12.2023 | 90,3600 | 89,4525 | 93,5296 | 88,8773 | +1.01% |
| 01.11.2023 | 89,4525 | 93,4000 | 93,7412 | 87,6858 | -4.23% |
| 01.10.2023 | 93,4000 | 97,9675 | 102,2742 | 91,6394 | -4.66% |
| 01.09.2023 | 97,9675 | 96,0250 | 98,5710 | 92,5561 | +2.02% |
| 01.08.2023 | 96,0250 | 91,6025 | 101,7356 | 91,4490 | +4.83% |
| 01.07.2023 | 91,6025 | 89,5500 | 93,7452 | 88,2854 | +2.29% |
| 01.06.2023 | 89,5500 | 81,1025 | 89,6863 | 80,4990 | +10.42% |
| 01.05.2023 | 81,1025 | 80,2500 | 81,3508 | 75,2510 | +1.06% |
| 01.04.2023 | 80,2500 | 77,5975 | 83,4458 | 77,3763 | +3.42% |
| 01.03.2023 | 77,5975 | 74,9700 | 77,9058 | 74,8825 | +3.50% |
| 01.02.2023 | 74,9700 | 69,8200 | 76,4350 | 69,8076 | +7.38% |
| 01.01.2023 | 69,8200 | 69,9000 | 72,4513 | 67,0998 | -0.11% |

Рисунок 19 - Входные данные

Данные после обработки:

| | |
|------------|-------|
| 2015-08-01 | 64,27 |
| 2015-09-01 | 65,45 |
| 2015-10-01 | 63,94 |
| 2015-11-01 | 66,48 |
| 2015-12-01 | 73,59 |
| 2016-01-01 | 75,46 |
| 2016-02-01 | 75,16 |
| 2016-03-01 | 67,02 |
| 2016-04-01 | 64,65 |
| 2016-05-01 | 66,69 |
| 2016-06-01 | 63,97 |

| | |
|------------|-------|
| 2016-07-01 | 66,04 |
| 2016-08-01 | 65,33 |
| 2016-09-01 | 62,83 |
| 2016-10-01 | 63,38 |
| 2016-11-01 | 64,14 |

Рисунок 20 - Данные после обработки

Анализ:

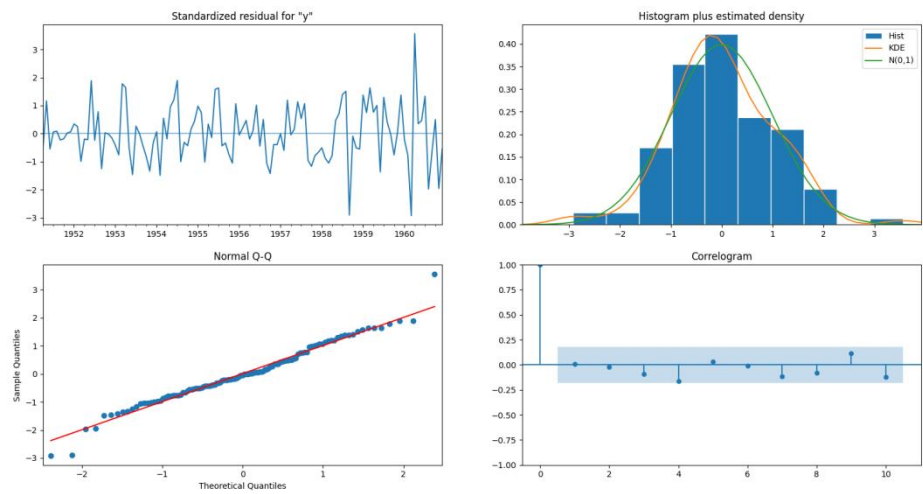


Рисунок 21 - Анализ ряда USD

Прогноз курса доллара:

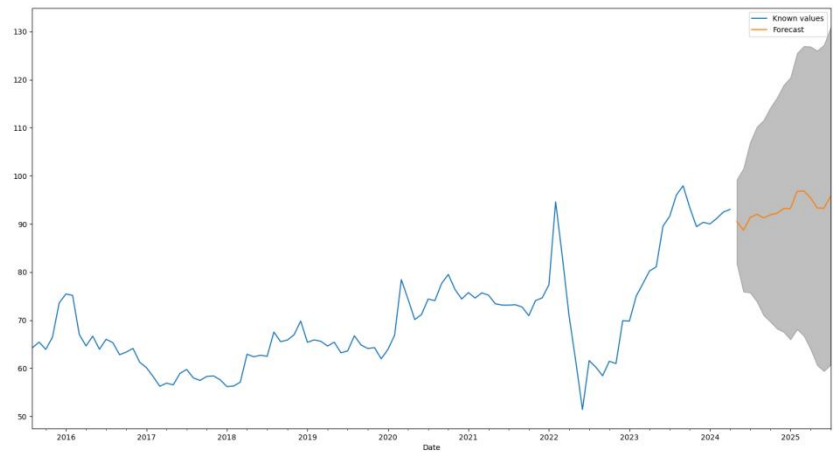


Рисунок 22 - Прогноз ряда USD

6.2 Временной ряд количества пассажиров, перевезенных авиакомпаниями

Входные данные:

| | |
|------------|-----|
| 1949-01-01 | 112 |
| 1949-02-01 | 118 |
| 1949-03-01 | 132 |
| 1949-04-01 | 129 |
| 1949-05-01 | 121 |
| 1949-06-01 | 135 |
| 1949-07-01 | 148 |
| 1949-08-01 | 148 |
| 1949-09-01 | 136 |
| 1949-10-01 | 119 |
| 1949-11-01 | 104 |
| 1949-12-01 | 118 |
| 1950-01-01 | 115 |

Рис. 23 - Входные данные

Ежемесячные данные о количестве пассажиров, перевезенных одной американской авиакомпанией с 1949 по 1960 годы:

Анализ:

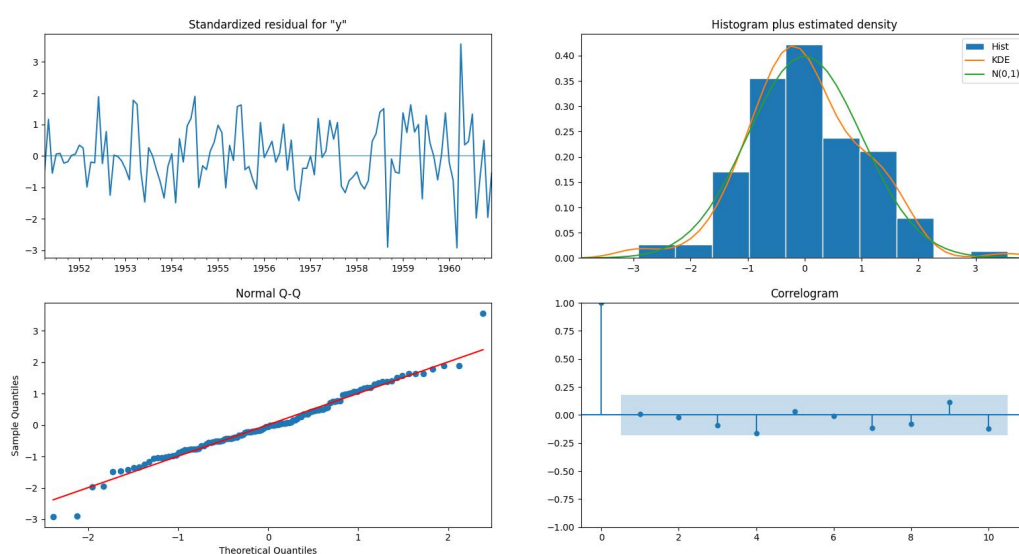


Рисунок 24 - Анализ ряда

Прогноз:

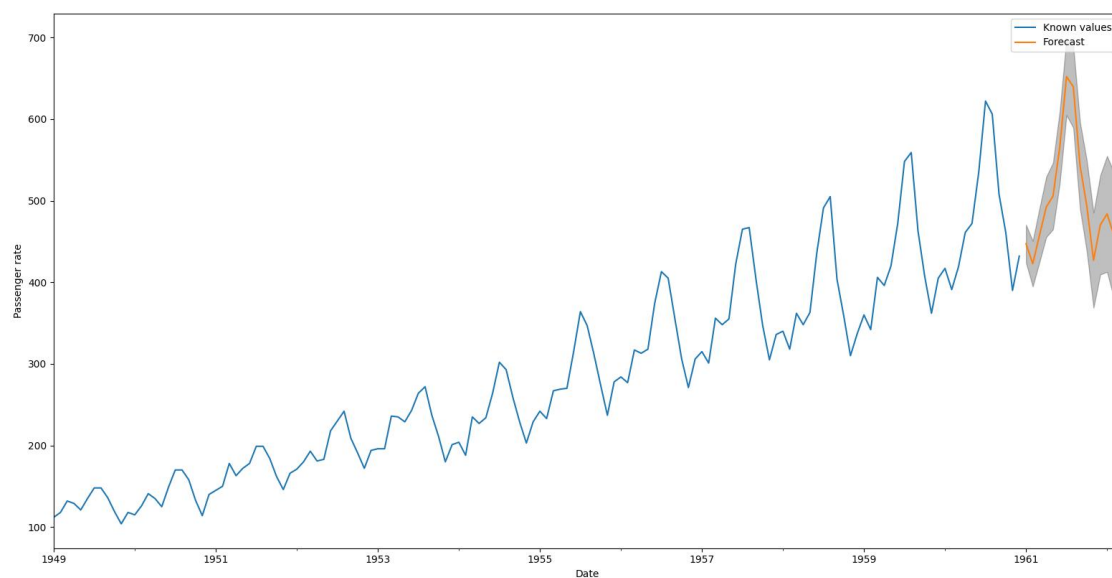


Рисунок 25 - Прогноз количества пассажиров

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.

ЛИСТИНГ ПРОГРАММЫ (МОДЕЛЬ ARIMA)

```
import warnings
import itertools
import pandas as pd
import statsmodels.api as sm
import matplotlib.pyplot as plt

# Данные CO2
data = sm.datasets.co2.load_pandas()
y = data.data
y = y['co2'].resample('MS').mean() # 'MS' группирует месячные данные
y = y.fillna(y.bfill())           # bfill значит, что нужно использовать
                                  # значение до заполнения пропущенных значений
print("\nДанные:\n", y, sep="")

# Задаю стиль графику и рисую его
# plt.style.use('fivethirtyeight')
y.plot(figsize=(15, 6))
plt.show()

# Определяю параметры p, d, q в диапазоне от [0,1]
p = d = q = range(0, 2)
print("\nЗначения (p, d, q):",
      "\np: ", p,
      "\nd: ", d,
      "\nq: ", q)

# Генерируем различные комбинации p, q и q
pdq = list(itertools.product(p, d, q))
print("\nВсе возможные комбинации (p, d, q):",
      "\n(p, d, q):", pdq, "\n")

# Сгенерируйте комбинации сезонных параметров p, q и q
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:\n",
      "SARIMAX: {} x {}".format(pdq[1], seasonal_pdq[1]), "\n",
      "SARIMAX: {} x {}".format(pdq[1], seasonal_pdq[2]), "\n",
      "SARIMAX: {} x {}".format(pdq[2], seasonal_pdq[3]), "\n",
      "SARIMAX: {} x {}".format(pdq[2], seasonal_pdq[4]), "\n", sep = "")
```

```

# Вычисление минимального (оптимального) AIC для ARIMA
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(dispatch=False)

            # log_likelihood = results.llf
            # k = len(results.params)
            # print('k = ', k)
            # print('aic = ', 2 * k - 2 * log_likelihood)

            print('ARIMA{0}x{1}12 - AIC:{0}'.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(dispatch=False)

# Вывод таблицы и описание её значений
print("\n", results.summary().tables[1])
print("\nar.L1          Коэффициент авторегрессии (AR)\n",
    "ma.L1             Коэффициент скользящего среднего (MA)\n",
    "ar.S.L12           Коэффициент сезонной авторегрессии (SAR)\n",
    "ma.S.L12           Коэффициент сезонного скользящего среднего (SMA)\n",
    "sigma2             Оценка дисперсии ошибок модели\n\n",
    "coef              Важность каждого параметра и его влияние на временной
ряд\n",
    "str err           Оценка стандартного отклонения коэффициента\n",
    "z                 Мера статистической значимости коэффициента модели\n",
    "P>|z|             Вероятность получения такого же или более экстремального
значения коэффициента\n",
    "[0.025...0.975]   Промежуток доверительного интервала\n", sep="")

# Рисуем 4 графика, которые показывают свойства временного ряда и его прогноза
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()
```

```
# Рисую график прогноза и действительного значения
```

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

```
pred_dynamic.predicted_mean.plot(label='Forecast', 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('Date')
```

```
ax.set_ylabel('CO2 rate') # BTC rate
```

```
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('Среднеквадратическая ошибка прогнозов равна: {}'.format(round(mse, 2)))
```

```
# Создание и визуализация прогноза. Получить прогноз на 15 шагов вперёд
```

```
pred_uc = results.get_forecast(steps=15)
```

```
# Получить интервал прогноза
```

```
pred_ci = pred_uc.conf_int()
```

```
# Рисую график прогноза
```

```
ax = y.plot(label='Known values', 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 rate') # BTC rate
```

```
plt.legend()
```

```
plt.show()
```

ЛИСТИНГ ПРОГРАММЫ (ОБРАБОТКА ДАННЫХ)

```
import java.io.IOException;
import java.io.FileWriter;
import java.util.Scanner;
import java.io.File;

public class Main {
    public static void main(String[] args) throws IOException {
        Scanner scan = new Scanner(new File("BTC_Weekly.txt"));
        FileWriter fw = new FileWriter(new File("BTC_Corrected_Weekly.txt"));

        /*Scanner scan = new Scanner(new File("BTC_Month.txt"));
        FileWriter fw = new FileWriter(new File("BTC_Corrected_Month.txt"));*/

        String str;
        String[] mass = new String[512];
        Integer strCount = 0;

        while (scan.hasNextLine()) {
            String ptr = scan.nextLine();

            String data = ptr.substring(0, ptr.indexOf("\t"));
            str = data.substring(data.lastIndexOf(".") + 1, data.length()) + "-";
// year
            str += data.substring(data.indexOf(".") + 1, data.lastIndexOf(".")) +
"-"; // month
            str += data.substring(0, data.indexOf(".")); // day

            ptr = ptr.substring(ptr.indexOf("\t") + 1, ptr.length());

            String value = ptr.substring(0, ptr.indexOf("\t") - 2);
            value = value.replace(".", "");

            str += "    " + value + "\n";
            mass[strCount] = str;
            strCount++;
        }
        System.out.println("Кол-во строк: " + strCount);

        for (int i = strCount - 1; i > 0 ; i--) {
            System.out.println(mass[i]);
            fw.write(mass[i]);
        }
        scan.close();
        fw.close();
    }
}
```