Оглавление

[1. Выявление характерных свойств набора данных для выбранной функции 2](#_Toc162202837)

[1.1. Предварительное знакомство с данными. 2](#_Toc162202838)

[2. Исследовательский анализ данных 6](#_Toc162202839)

[2.1. Возможные признаки проявления сезонности 6](#_Toc162202840)

[3. Моделирование «сезонности» 11](#_Toc162202841)

[3.1. Равномерное агрегирование по времени 11](#_Toc162202842)

[3.2. Сдвиг по параметру 12](#_Toc162202843)

[3.3. Сравнение моделей 13](#_Toc162202844)

[3.4. Параметрическая модель «сезонности» 14](#_Toc162202845)

[3.5. Постановка задачи 15](#_Toc162202846)

[4. Кластерный анализ параметрической модели сезонности 16](#_Toc162202847)

[4.1. Построение центроида на множестве моделей 16](#_Toc162202848)

[4.2. Итерационная процедура получения центроида с характерным свойством набора данных 18](#_Toc162202849)

[Результат 22](#_Toc162202850)

[Заключение 24](#_Toc162202851)

[Приложение 26](#_Toc162202852)

[Описание функций 26](#_Toc162202853)

# Выявление характерных свойств набора данных для выбранной функции

В ходе работы рассматривается набор данных, который представляет из себя временной ряд значений коэффициента опредмеченности действия в наблюдаемый период с 21 декабря 2019 года по 05 марта 2023 года (всего 1171 значение).

Необходимо определить характерные признаки проявления «сезонности» в наборе данных за период наблюдений.

В отчете представлен рабочий процесс выявления характерных свойств набора данных для выбранной функции, реализованный в Python.

## Предварительное знакомство с данными.

На этапе предварительного знакомства необходимо получить первоначальное представление о наборе данных для определения последовательности дальнейших шагов.

Для начала нужно понять, нуждаются ли данные в предварительной обработке, то есть проверить наличие пропусков в данных, и понять, нужна ли очистка данных. Затем необходимо визуализировать данные, чтобы установить возможный характер распределения и выяснить главные особенности набора по описательным статистикам.

С использованием библиотеки pandas были загружены данные из файла coef\_opr\_actions.xlsx. Данные оказались без пустых значений, и одного типа, то есть очистка не требуется.

С помощью функции describe() получаю описательные характеристики данных.

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 1 - Вывод функции describe()

Функция вернула количество данных; среднее значение; стандартное отклонение; минимальное значение; первый, второй и третий квартили; максимальное значение.

Далее, используя matplotlib визуализированы данные на графике.

Изображение выглядит как снимок экрана, График, линия

Автоматически созданное описание

Рисунок 2 – Визуализация данных на графике

Далее строю диаграмму распределения данных

Изображение выглядит как диаграмма, График, снимок экрана, линия

Автоматически созданное описание

Рисунок 3 - Диаграмма распределения данных

Чтобы увидеть на графике потенциальные выбросы, необходимо визуализировать области стандартного отклонения.

Изображение выглядит как текст, снимок экрана, График, линия

Автоматически созданное описание

Рисунок 4 - Визуализация областей Std

По графику можно заметить, что большинство данных входит в область стандартного отклонения. Несколько значений выходят за пределы трех стандартных отклонений, такие данные скорее всего являются выбросом.

Также дополнительно создается таблица, показывающая количество данных и накопление по областям.

После получаю процент данных, входящих в каждую область и накопление по областям

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 5 - Подсчет данных по областям

На основе полученных данных можно увидеть, что 6 строк являются потенциальным выбросом. Последнее значение временного ряда (запись от 05 марта 2023 года), расположенное в области «> 3 Std» (т. е. вне пределов 3-х сигма), вероятно является техническим выбросом, хотя, возможно, и аномальным значением. В данном отчете не исследуются выбросы и аномалии, и так как это последнее значение в наборе данных, его можно отбросить без нарушения целостности набора данных и далее не рассматривать.

**Вывод: д**анные не содержат пропусков, очистка не требуется, распределение близко к нормальному и хорошо согласуется с эмпирическим правилом «трех сигм», в дальнейшем целесообразно исключить из набора последнее значение и рассматривать 1170 записей с 21 декабря 2019 года по 04 марта 2023 года.

# Исследовательский анализ данных

Исследовательский анализ данных (EDA, Exploratory Data Analysis) – это второй этап рабочего процесса по выявлению характерных свойств набора данных, на котором нужно выбрать «функцию» (feature), и по ней определить признаки проявления «сезонности» в наборе данных за период наблюдений.

Из таблицы на рис. 5 видно, что более 95 % данных расположено в пределах 2-х сигма, и именно эти значения определяют основные свойства набора данных. Значения, которые расположены вне этих пределов, не оказывают существенного влияния на проявление основных свойств всего набора.

## Возможные признаки проявления сезонности

Признаки сезонности могут проявляться в характере распределения данных по двум областям: «1 Std» и « > 1 Std». Одной из функций, с помощью которой возможно описать характер распределения данных по указанным областям, является процентное соотношение количества данных, содержащихся в этих областях.

В процессе анализа данных были определены две категории: белые и черные дни. Белые дни соответствуют дням, входящим в область стандартного отклонения, то есть 1 Std, в то время как черные дни представляют собой остальные дни набора данных. Для классификации дней по этим категориям была реализована функция, добавляющая соответствующие метки к данным. После этого было определено базовое распределение черных и белых дней по всему набору данных. Базовый уровень представляет собой вектор из двух компонентов, где:

BL1 = (100 \* "Кол-во данных в области 1 Std" / "Общее кол-во данных")

BL2 = (100 \* "Кол-во данных в области > 1 Std" / "Общее кол-во данных")

Для выявления признаков наличия сезонности в данных, данные были агрегированы по временам года за все годы наблюдений. Была реализована функция, позволяющая получить таблицу с уровнем белых и черных дней по датасету, а также отклонение от базового уровня набора данных. Для агрегации данных по сезонам были выбраны следующие временные интервалы:

Весна: с 01 марта по 31 мая;

Лето: с 01 июня по 31 августа;

Осень: с 01 сентября по 30 ноября;

Зима: с 01 декабря по 28 февраля.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 6 - Таблица уровней белых и черных дней по временам года

Для упрощения восприятия данная таблица была визуализирована в виде графика

Изображение выглядит как текст, диаграмма, снимок экрана, Прямоугольник

Автоматически созданное описание

Рисунок 7 - Визуализация уровней белых и черных дней по временам года

После определения базового распределения белых и черных дней по всему набору данных. Для более детального изучения сезонности, данные были дополнительно агрегированы по квартальным интервалам за все годы наблюдений. Для этого была реализована функция, аналогичная той, которая использовалась для агрегации данных по сезонам. Данные были выделены по следующим временным интервалам:

1-й квартал: с января по март включительно;

2-й квартал: с апреля по июнь включительно;

3-й квартал: с июля по сентябрь включительно;

4-й квартал: с октября по декабрь включительно.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 8 - Таблица уровней белых и черных дней по кварталам

Далее данные были визуализированы.

Изображение выглядит как диаграмма, текст, снимок экрана, прямоугольный

Автоматически созданное описание

Рисунок 9 - Визуализация уровней белых и черных дней по кварталам

После агрегации данных по квартальным интервалам, данные были дополнительно агрегированы по межсезонным интервалам, что позволило более детально изучить переходные периоды между сезонами. Для этого были выбраны следующие временные интервалы:

"Февраль-Апрель" - с февраля по апрель включительно;

"Май-Июль" - с мая по июль включительно;

"Август-Октябрь" - с августа по октябрь включительно;

"Ноябрь-Январь" - с ноября по январь включительно;

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 10 - Таблица уровней белых и черных дней в межсезонье

Изображение выглядит как текст, диаграмма, снимок экрана, Прямоугольник

Автоматически созданное описание

Рисунок 11 - Визуализация уровней белых и черных дней в межсезонье

**Вывод:** в ходе исследовательского анализа данных были выявлены ключевые характеристики набора данных, включая распределение значений по областям стандартного отклонения и возможные признаки сезонности. Было установлено, что более 95% данных располагаются в пределах двух сигм, что указывает на нормальное распределение данных. Признаки сезонности были выявлены в распределении данных по категориям белые и черные дни. Отклонения от базового уровня белых и черных дней варьируются в зависимости от времени года и кварталов, что подчеркивает динамичность сезонных колебаний.

Можно заметить, что в зимний период наблюдается снижение активности, что может быть связано с сезонными изменениями в поведении или деятельности, отраженной в данных. Весенний период характеризуется изменением сезонных тенденций, что может указывать на начало нового активного периода. Летом наблюдается пик активности, что может быть связано с максимальным уровнем сезонного влияния на данные. Осенний период характеризуется конечным снижением активности, что может указывать на завершение активного сезона.

В квартальном анализе также наблюдаются сезонные колебания, хотя они менее выражены по сравнению с годовыми сезонами. Это может указывать на более сложную динамику сезонности, которая может быть связана с внутренними сезонными циклами в данных. В межсезонье уровни белых и черных дней варьируются, что может указывать на периоды перехода между сезонами.

# Моделирование «сезонности»

Модели «сезонности» определяются периодом и параметром сдвига. Для анализа данных используются модели, определяющие ключевые параметры, такие как уровни белого и черного.

## 3.1. Равномерное агрегирование по времени

Для рассматриваемого набора данных лучше подходит модель с равномерным агрегированием по времени с периодом 90 дней. Данная модель позволяет исследовать характерные особенности в распределении данных по соотношениям уровней белых и черных дней в выбранные периоды.

Всего в наборе 1170 дней. Для интервала в 90 дней получаем 1170 / 90 = 13, т.е. в наборе данных будет 13 интервалов по 90 дней. Далее для полученной модели определяются уровни белого и черного и отклонение от базового уровня.

Изображение выглядит как текст, снимок экрана, меню

Автоматически созданное описание

Рисунок 12 - таблица уровней черных и белых дней для равномерно агрегированной модели

Изображение выглядит как текст, музыка

Автоматически созданное описание

Рисунок 13 - визуализация уровней черных и белых дней для равномерно агрегированной модели

## 3.2. Сдвиг по параметру

Параметр сдвига определяет «положение» модели относительно набора данных. Изменение параметра сдвига влияет на характер «рисунка» в соотношениях уровней белых и черных дней. Разным значениям параметра сдвига соответствуют разные модели. Положительное значение параметра сдвига означает сдвиг модели на одну единицу влево, а отрицательное вправо, то есть на l дней назад или вперед. При этом, при сдвиге модели вправо размер первой ячейки увеличится на l дней, а размер последней уменьшится на l, и наоборот, при сдвиге влево.

Реализована функция shift\_model(), позволяющая получить модель, задавая только значение параметра сдвига l.

## 3.3. Сравнение моделей

Модели с различными значениями параметра сдвига сравниваются по «положению» в метрике соответствия данным и по «значению» в метрике R2.

Для l = 1, 0 и -1 строятся три модели, после чего определяются уровни белого и черного для каждой модели, а затем находится fit\_error, как евклидова норма вектора разности [BW - BL] = [W - BL1, B - BL2]. Это позволяет оценить, как изменение параметра сдвига влияет на модель. Ниже изображены выводы функций с оценками моделей.



Рисунок 14 - Уровень черного и белого для модели l = 0



Рисунок 15 - fit\_error для модели l = 0



Рисунок 16 - Уровень черного и белого для модели l = 1

Изображение выглядит как Шрифт, текст, Графика, типография

Автоматически созданное описание

Рисунок 17 - fit\_error для модели l = 1



Рисунок 18 - Уровень черного и белого для модели l = -1

Изображение выглядит как текст, Шрифт, Графика, белый

Автоматически созданное описание

Рисунок 19 - fit\_error для модели l = -1

Далее модели сравниваются между собой с использованием метрики R-квадрат, где ytrue представляет модель с параметром l = 0, а ypred - модели для сравнения. Расстояние между моделями с параметрами l = 1 и l = -1 вычисляется как 1 – R^2. Это позволяет оценить, насколько модели с разными параметрами сдвига соответствуют данным.

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

Рисунок 20 - расстояние между моделями

Все полученные оценки объединяются в одну таблицу для наглядности. Таблица позволяет легко сравнивать результаты различных моделей и выбирать наиболее подходящую для анализа данных. Этот подход обеспечивает детальное понимание характеристик данных и эффективности моделей, что помогает принимать решения на основе полученных результатов.

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

Рисунок 21 - Таблица fit\_error и расстояния

## 3.4. Параметрическая модель «сезонности»

Параметрическое семейство равномерно-агрегированных моделей при варьировании параметра сдвига в определенных пределах «покрывает» весь набор данных и формирует параметрическую модель «сезонности» с выбранным периодом. В такой параметрической модели реализуются все возможные «рисунки» соотношений уровней белых и черных дней. Изображение выглядит как График, снимок экрана, линия, диаграмма

Автоматически созданное описание

Рисунок 22 - график зависимости fit-error от значения параметра l

## 3.5. Постановка задачи

Для параметрической модели «сезонности» выделить характерные устойчивые сочетания «рисунков» с близкими свойствами, которые наилучшим образом соответствуют данным.

Для решения поставленной задачи применяется подход, основанный на идеях кластеризации.

# Кластерный анализ параметрической модели сезонности

Кластерный анализ параметрической модели сезонности направлен на выявление характерного свойства набора данных. На первом этапе – выявлении характерного рисунка по «значению» в метрике R2 – применяется техника, основанная на построении центроидов.

Для выявления характеристик набора данных центроиды представляют собой эффективный метод анализа. Центроид представляет собой агрегированный объект, усредненный по нескольким моделям, и определяется как среднее значение уровней "белого" и "черного" по всем моделям в рассматриваемом множестве. Процесс построения центроидов и их применение представляются важными шагами анализа данных. Центроид представляет собой вектор, состоящий из k значений, полученных путем усреднения соответствующих значений отдельных моделей в множестве. По агрегированным значениям центроида также можно определить совокупный уровень "белого" и "черного", что представляет собой вектор усредненных значений этих параметров.

## 4.1. Построение центроида на множестве моделей

Построение центроида на множестве моделей начинается с агрегации данных из имеющегося множества моделей, включающего 81 модель с параметром l, изменяющимся в пределах от -40 до 40. Каждая модель определяется двумя векторами, состоящими из k = 13 чисел, представляющими уровни "белого" и "черного" соответственно. Построенный центроид представляет собой два вектора, характеризующих уровни "белого" и "черного".

Изображение выглядит как текст, Шрифт, снимок экрана, типография

Автоматически созданное описание

Рисунок 23 - векторы уровней белого и черного

Центроид является усредненной моделью, обобщающей свойства всех тех моделей, на множестве которых он был построен. Функция, известная как fit-error и уже описанная в предыдущих разделах, вычисляет его «положение» в метрике соответствия данным.

Изображение выглядит как текст, Шрифт, Графика, логотип

Автоматически созданное описание

Рисунок 24 - значение fit-error для модели с параметром l = 0

Затем для каждой модели из рассматриваемого множества вычисляются расстояния до базовой модели с параметром l = 0. Эти расстояния выражены в метрике R^2 и объединены в одну таблицу для удобства анализа. Также оценивается соответствие каждой модели центроиду, используя метрику R^2 без коррекции на базовую модель.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 25 - таблица расстояний для модели с параметром l = 0

## 4.2. Итерационная процедура получения центроида с характерным свойством набора данных

Итерационная процедура получения центроида с характерным свойством набора данных используется для выявления обобщенной модели, которая наилучшим образом соответствует характеристикам исследуемого набора данных.

На этапе инициализации (нулевом шаге) в качестве начального выбирается центроид, построенный на всем множестве моделей параметрического семейства.

На каждом следующем шаге отсеиваются модели, расстояние которых по «значению» в метрике R2 до центроида больше среднего. Оставшиеся модели образуют множество для построения нового центроида на этом шаге.

Итерационная процедура останавливается, когда перестает изменяться множество моделей для построения очередного центроида.

В результате итерационной процедуры получается обобщенная модель, представленная в виде центроида, которая описывает ключевые характеристики данных и может быть использована для дальнейшего анализа и прогнозирования.

Определено среднее значение в столбце R^2 to centroid, данное значение является порогом, то есть threshold.

Изображение выглядит как Шрифт, текст, Графика, белый

Автоматически созданное описание

Рисунок 26 - значение threshold

Дальше из всего множества моделей (81 штука) отбираются те модели, где соответствие центроиду в метрике *R*2 превышает порог. На первой итерации центроид включает в себя 51 модель.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 27 - первые 10 из отобранных моделей

Изображение выглядит как текст, Шрифт, снимок экрана, черно-белый

Автоматически созданное описание

Рисунок 28 - центроид первой итерации

Определен fit\_error для нового центроида первой итерации. Полученное значение fit-error = 0.026. Далее определены значения R^2 to new centroid

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 29 - таблица расстояний и ошибка для центроида первой итерации  
Изображение выглядит как пианино, клавиатура, музыка, Электронный инструмент

Автоматически созданное описание

Рисунок 30 - визуализация центроида первой итерации

Далее проведена вторая итерация центроида.

Определено новое значения threshold, равное 0.9479981481481482

Отобраны модели из моделей, оставшихся после первой итерации центроида (54 штуки), далее сравниваем R^2 to new centroid с новым значением threshold, и получаем 34 модели. Аналогично строим центроид второй итерации, то есть считаем векторы Wc и Bc по отобранным моделям. Получены значения

Изображение выглядит как текст, Шрифт, снимок экрана, черно-белый

Автоматически созданное описание

Рисунок 31 - вторая итерация центроида

Затем производится оценка соответствия центроида данным, получаем значение fit-error = 0.0124.

Процедура продолжается с установкой нового порогового значения и выбором подходящих моделей. Этот процесс повторяется до тех пор, пока не будут рассмотрены все доступные модели. В случае представленных данных получилось 6 итераций.

В процессе формирования центроидов происходит запись значений ошибки соответствия центроида для каждой итерации в вектор.



Рисунок 32 - вектор значений ошибки соответствия центроида для всех итерацийИзображение выглядит как линия, График, диаграмма, число

Автоматически созданное описание

Рисунок 33 - график зависимости ошибки от номера итерации

# Результат

Характерный рисунок набора данных, выраженный в соотношении уровней белых и черных дней по периодам в 90 дней.

Изображение выглядит как текст, музыка

Автоматически созданное описание

Рисунок 34 - характерный рисунок набора данных, выраженный в соотношении уровней белых и черных дней по периодам в 90 дней

Диаграммы центроида последней (шестой) итерации. Изображение выглядит как текст, пианино, музыка

Автоматически созданное описание

Рисунок 35 - диаграмма центроида последней итерации

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 36 - диаграмма центроида последней итерации (2)

# Заключение

В ходе проведения работы были выполнены основные этапы анализа данных и моделирования сезонности. Исследовательский анализ данных (EDA) позволил выявить основные характеристики набора данных, включая распределение значений и признаки сезонности. Моделирование "сезонности" проводилось с использованием моделей с равномерным агрегированием по времени и параметрической модели, что позволило выделить характерные особенности распределения данных. Выявление характерного свойства набора данных осуществлялось с помощью итерационной процедуры получения центроида, что позволило создать обобщенные модели, максимально соответствующие особенностям данных.

Второй этап кластерного анализа параметрической модели сезонности – локализация выявленного свойства – в этом отчете не рассматривается.

Полученные результаты представляют собой базу для дальнейшего анализа временных рядов, выявления закономерностей и прогнозирования будущих значений.

# Приложение

Код программы на Python выложен на [GitHub](https://github.com/nkt1lyn/data_analysis)

## Описание функций

|  |  |
| --- | --- |
| Функция | Как и что делает |
| data\_load() | Загружает данные из файла Excel, расположенного по указанному пути, и возвращает их в виде объекта Pandas DataFrame. |
| data\_shape(df, date1, date2) | Фильтрует DataFrame df, оставляя только строки, где столбец 'Time' находится в диапазоне дат от date1 до date2 (включительно). |
| shift\_model(df, l) | Сдвигает столбец 'Период' в DataFrame df на l позиций. Отрицательные значения l сдвигают вверх, положительные - вниз. Обрабатывает крайние случаи, заполняя сдвинутые значения 1 или 13. |
| day\_cat(df) | Классифицирует дни в DataFrame df как "Белые" или "Черные" на основе значений 'Var1'. "Белые дни" - это дни, где 'Var1' находится в пределах одного стандартного отклонения от среднего, "Черные дни" - все остальные. |
| model\_with\_shift(l) | Создает модель со сдвинутым столбцом 'Период' и категоризацией дней. Выполняет загрузку данных, фильтрацию по датам, категоризацию дней, создание столбца 'Период' и его сдвиг. |
| calculate\_levels(group) | Вычисляет процент "Белых" и "Черных" дней в группе строк DataFrame (обычно сгруппированных по 'Период'). |
| BW\_calc(df) | Вычисляет средние проценты "Белых" и "Черных" дней по всем периодам в DataFrame df. |
| plot\_shift(df, l) | Создает гистограмму, визуализирующую проценты "Белых" и "Черных" дней для каждого 'Периода' в DataFrame df после применения сдвига l. |
| fit\_error\_calc(BW, BL) | Вычисляет ошибку соответствия между рассчитанными уровнями BW и базовыми уровнями BL, используя евклидову норму. |
| FE\_table(list) | Создает DataFrame с ошибками соответствия для моделей с разными сдвигами. |
| WB\_list(models) | Извлекает проценты белых и черных дней из словаря моделей. |
| WсBc(W\_list, B\_list) | Вычисляет средние проценты белых и черных дней по всем моделям во входных списках. |
| BWc\_list(Wc, Bc) | Вычисляет среднее значение средних процентов белых и черных дней. |
| fit\_error\_c(BWc, BL) | Вычисляет ошибку соответствия между средними уровнями центроида и базовыми уровнями. |
| quarter\_label(df, year\_col=None) | Добавляет столбец 'Период' в DataFrame df, указывающий квартал года на основе месяца в столбце 'Time'. |
| season\_label(df, year\_col=None) | Добавляет столбец 'Период' в DataFrame df, указывающий время года на основе месяца в столбце 'Time'. |
| BW\_level(df, period, BaseLevel) | Вычисляет уровни белых и черных дней и отклонения для заданных периодов в DataFrame df. |
| BW\_level\_plot(level\_df, title) | Создает гистограмму, визуализирующую уровни белых и черных дней для каждого периода в DataFrame level\_df. |