

# Прикладные задачи на временных рядах и методы их решения

М. Габдушев, ETNA

А. Махин, ETNA

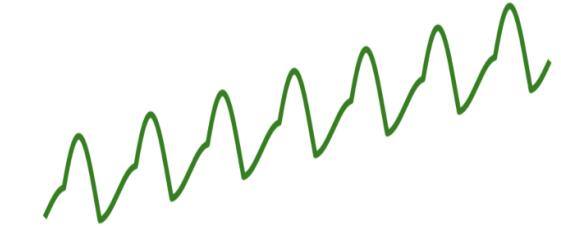
2023



# **Задачи на временных рядах в реальной жизни**

# Что такое временной ряд?

Временной  
ряд



Тренд

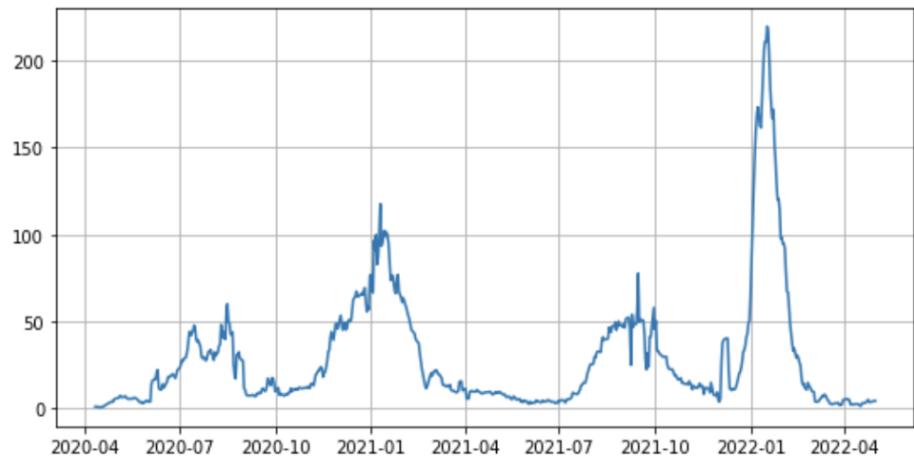


Сезонность



# Реальные примеры

Заболевания COVID в одном из городов США



КАК ЗАЕМЩИКИ ЖАЛОВАЛИСЬ НА КОЛЛЕКТОРОВ В 2021 ГОДУ источник: ФССП.



Доллар/Рубль USD/RUB  
₽61,880 (-2,47%)



# Задачи на временных рядах

- Прогнозирование
- Классификация
- Кластеризация
- Сегментация
- Поиск аномалий



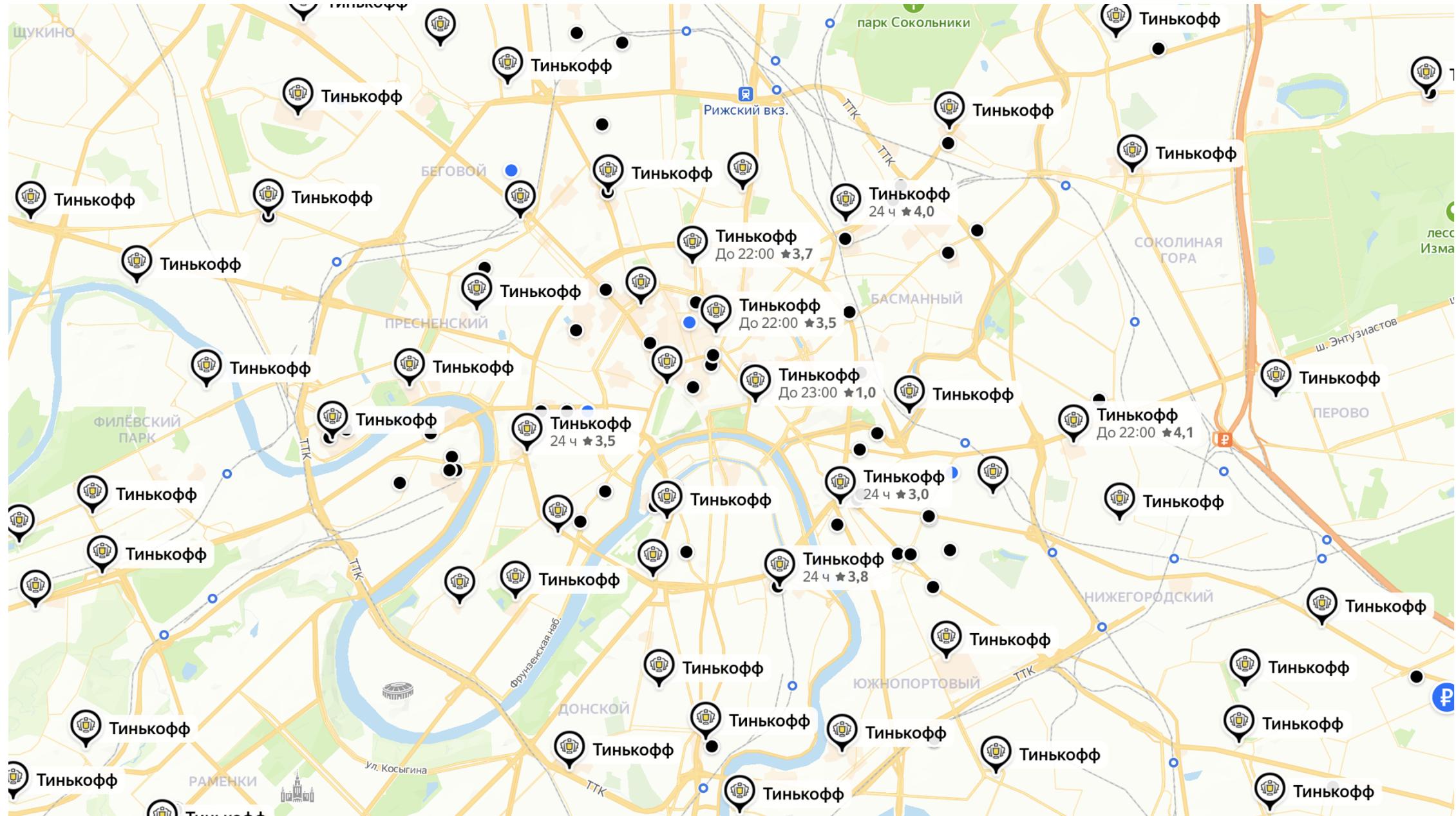
Планирование  
ресурсов

Для чего их нужно  
предсказывать?



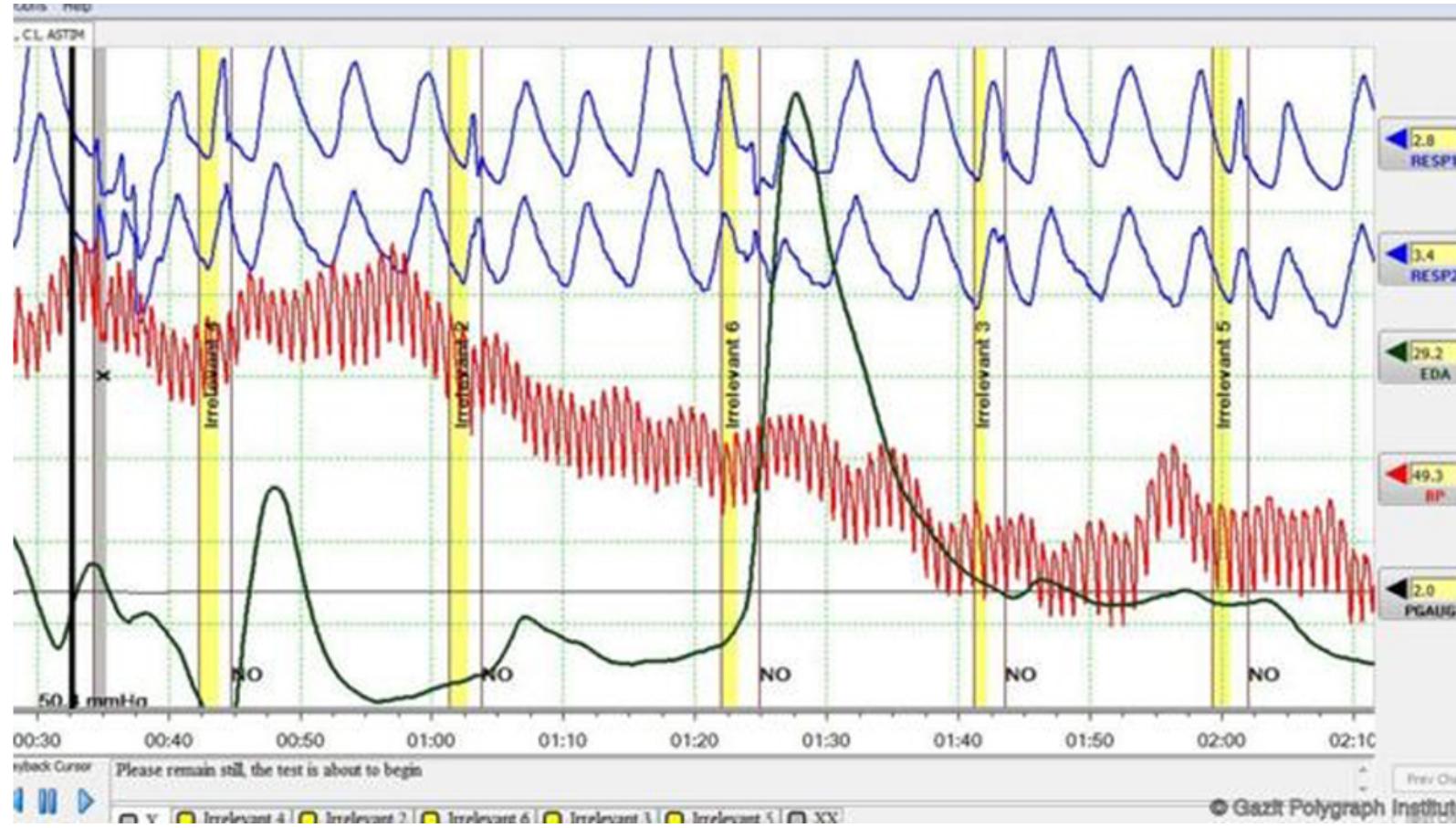
Планирование спроса  
и цепочек поставок





# Задачи на временных рядах

- Прогнозирование
- Классификация
- Кластеризация
- Сегментация
- Поиск аномалий



## Показатели с датчиков детектора лжи

- Врет?
- Говорит правду?

# Классификация

Где может быть полезна:

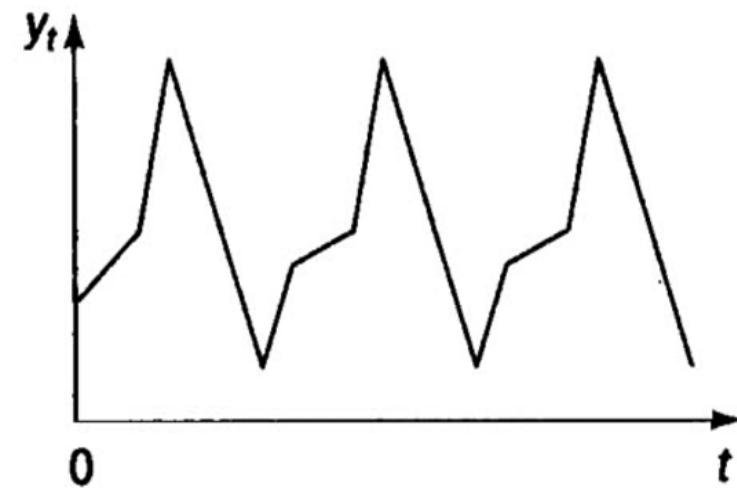
- Классификация пользователей
  - Фрод / не фрод
- Классификация активностей
  - По данным со смарт часов предсказать тип активности
- Детектор лжи

## Базовые подходы

Задаем метрику между рядами

$$d : X \times X \rightarrow \mathbb{R}_{\geq 0}$$

Применяем метрические методы  
классификации (например, kNN)

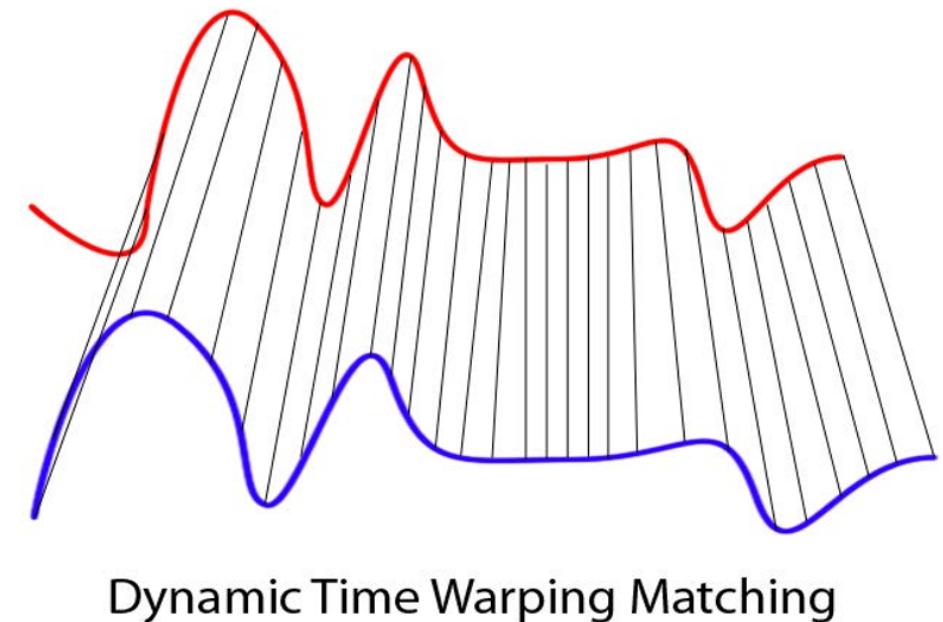
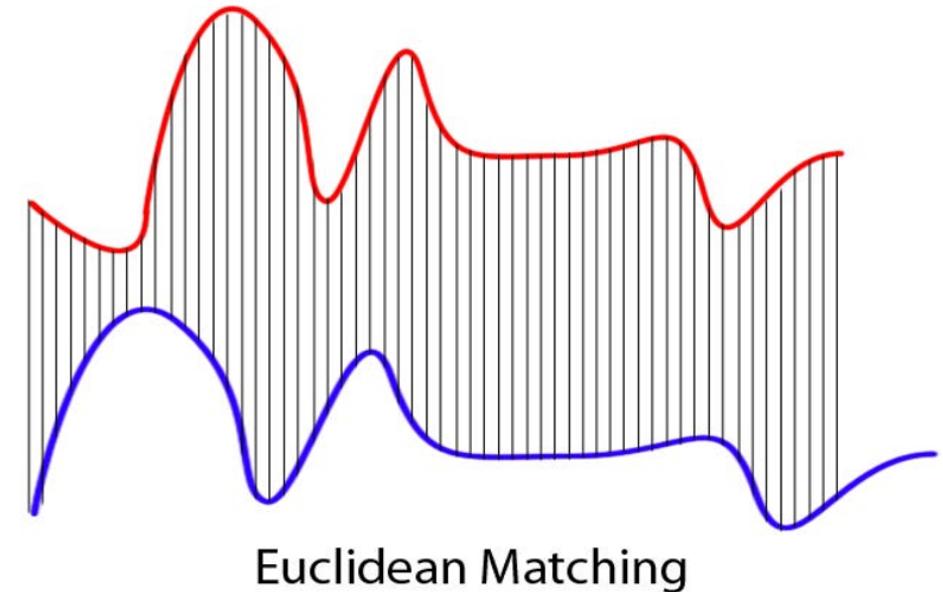


# DTW distance

Сравниваем два временных ряда

$$Q = (q_1, q_2, \dots, q_n) \text{ и}$$

$$R = (r_1, r_2, \dots, r_m)$$



[1] Dynamic time warping

## DTW distance

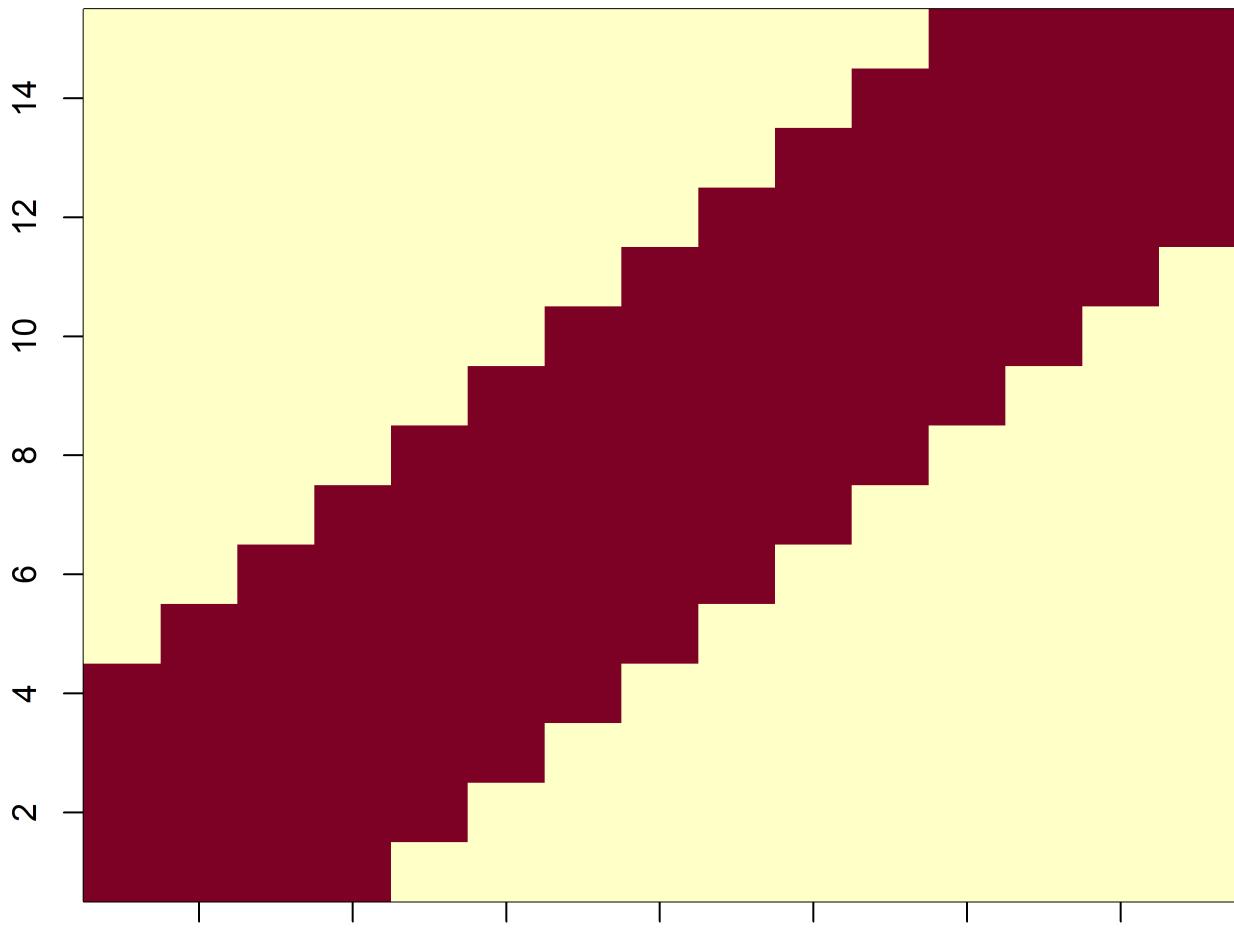
Сравниваем два временных ряда  $Q = (q_1, q_2, \dots, q_n)$  и  $R = (r_1, r_2, \dots, r_m)$

Составляем матрицу локальных потерь размера  $n \times m$ :  $lcm(i, j)$  содержит расстояние между парой наблюдений  $q_i$  и  $r_j$ .

Находим путь трансформации  $\phi = \{(1, 1), \dots, (n, m)\}$  через эту матрицу, который минимизирует суммарное расстояние между рядами.

## DTW distance

Чаще всего накладывают ограничение на монотонность пути и на ширину окна трансформации, в пределах которого разрешается прокладывать путь трансформации.



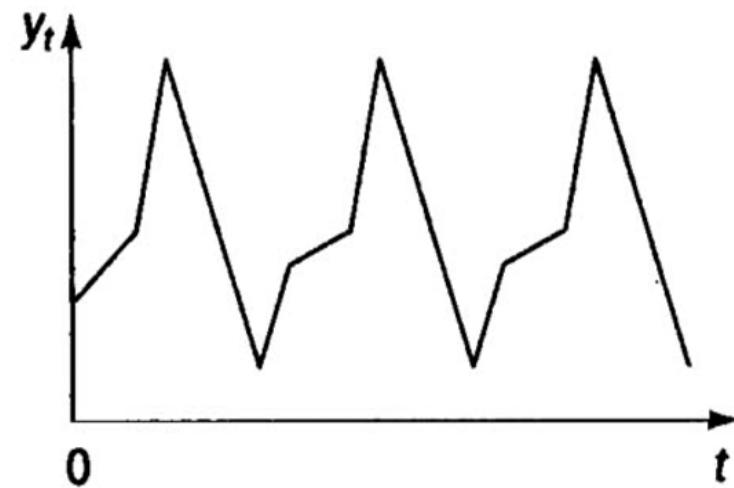
Query: samples 1..15

## Базовые подходы

Приводим ряд к табличным данным

- Статистики ряда
- Коэффициенты Фурье
- Признаки из моделей

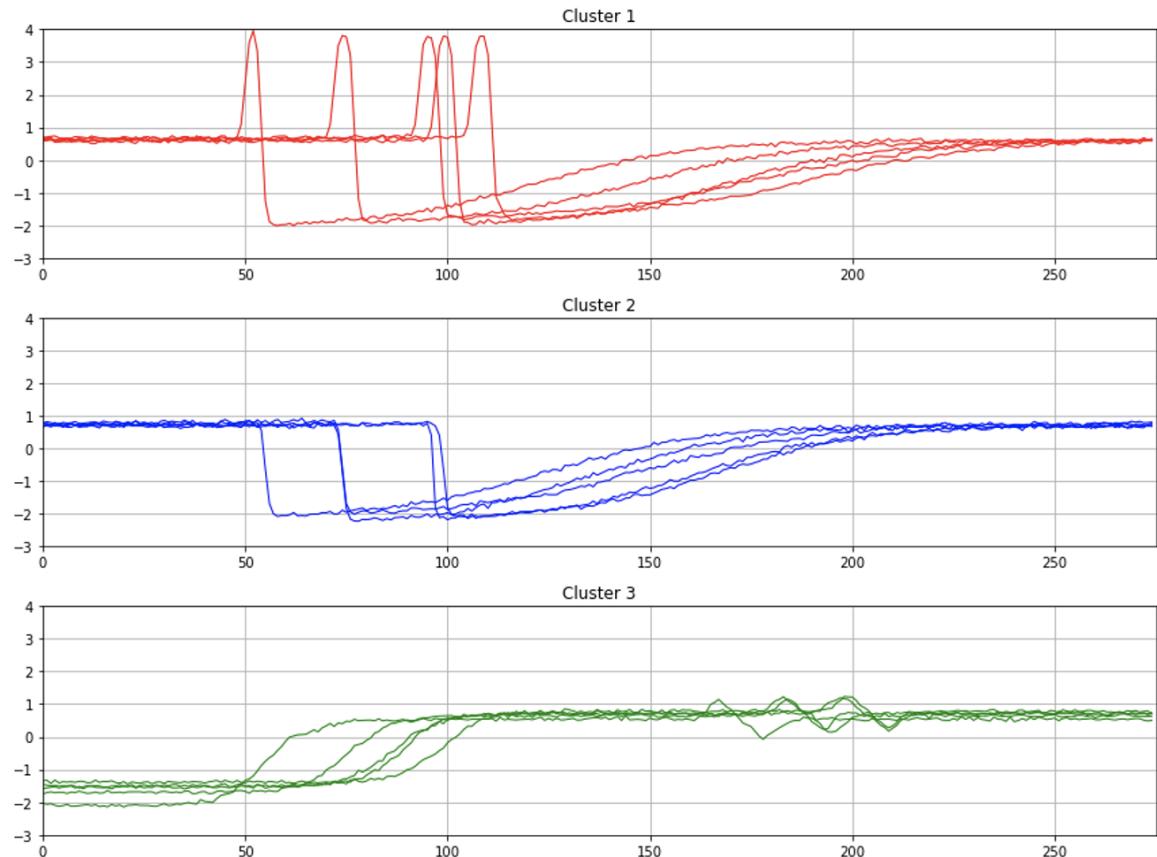
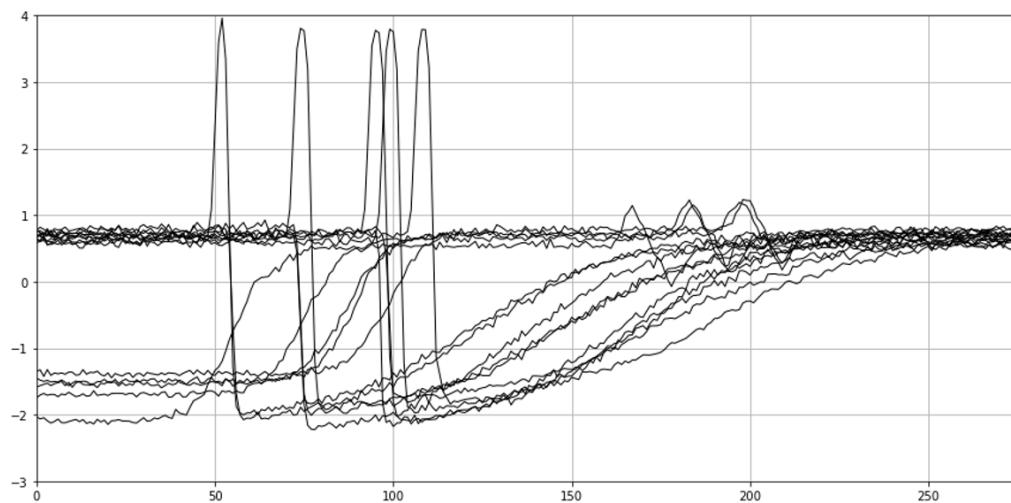
Применяем любые методы для  
классификации табличных данных



## Задачи на временных рядах

- Прогнозирование
- Классификация
- Кластеризация
- Сегментация
- Поиск аномалий

# Кластеризация



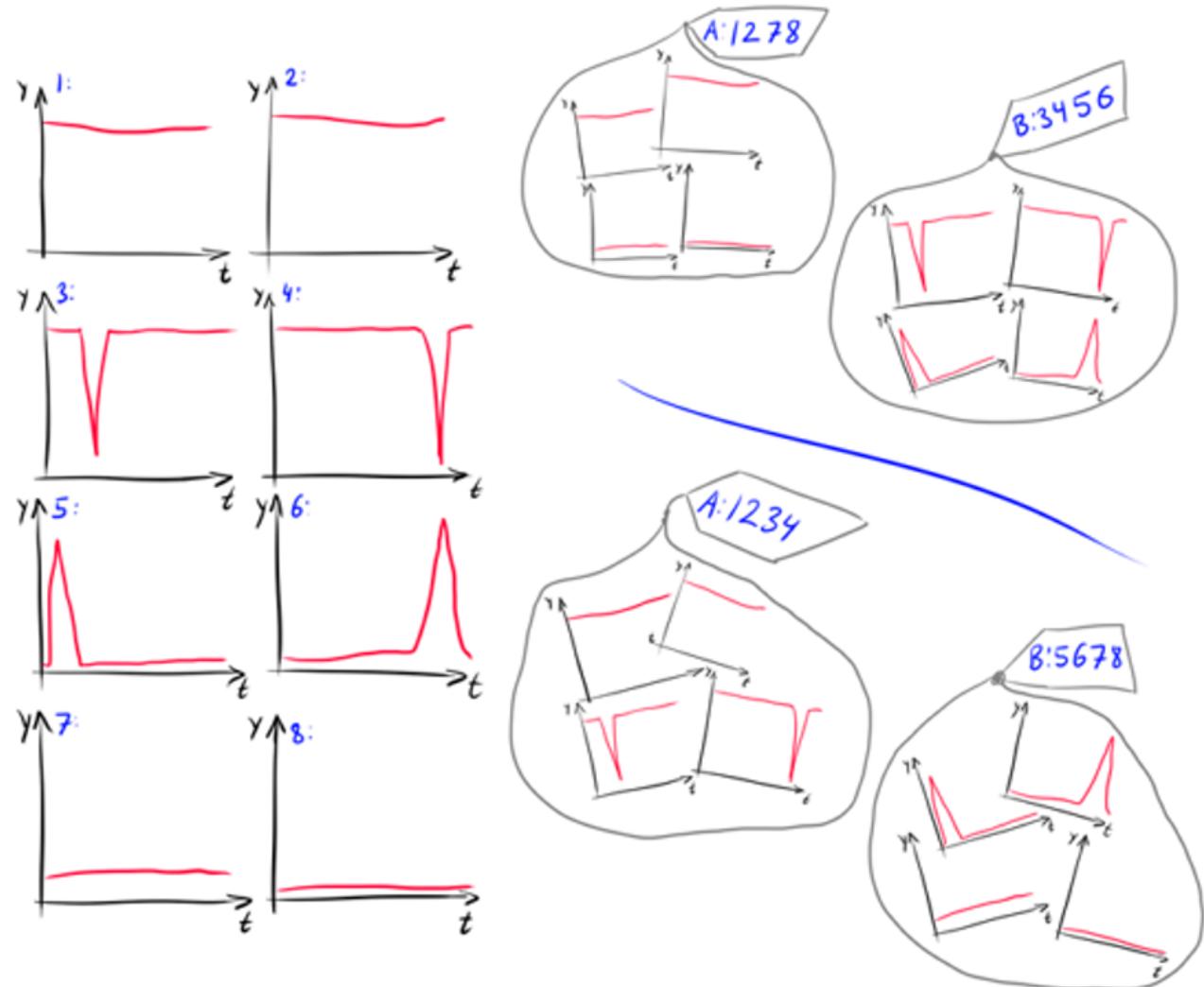
[1] Приёмы и метрики для кластеризации временных рядов

# Кластеризация

Где может быть полезна:

- Поиск похожих пользователей
  - К примеру, чтобы предлагать определенные условия
- В прогнозировании
  - Добавление информации в модель как метку кластера
  - Работа с каждым кластером отдельно
- Сокращение потребления ресурсов
  - Сохранение только одного ряда (центроиду) на каждый кластер

# Как правильно разделять на классы?



*Найдите лишнюю фигуру*

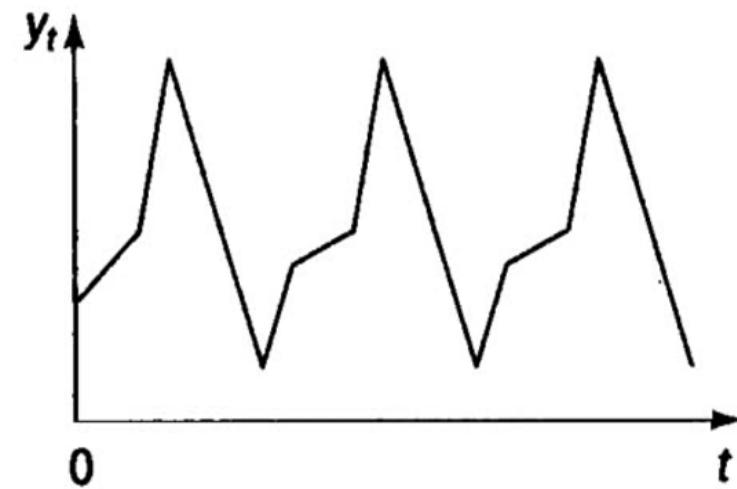


## Базовые подходы

Задаем метрику между рядами

$$d : X \times X \rightarrow \mathbb{R}_{\geq 0}$$

Применяем метрические методы  
кластеризации

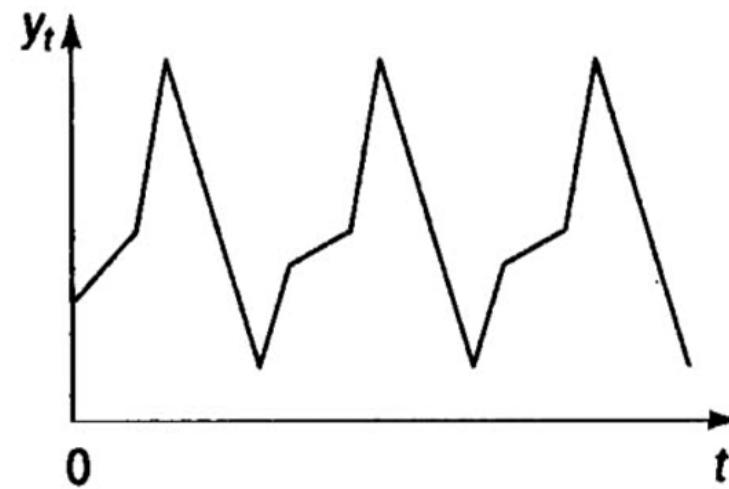


## Базовые подходы

Приводим ряд к табличным данным

- Статистики ряда
- Коэффициенты Фурье
- Признаки из моделей

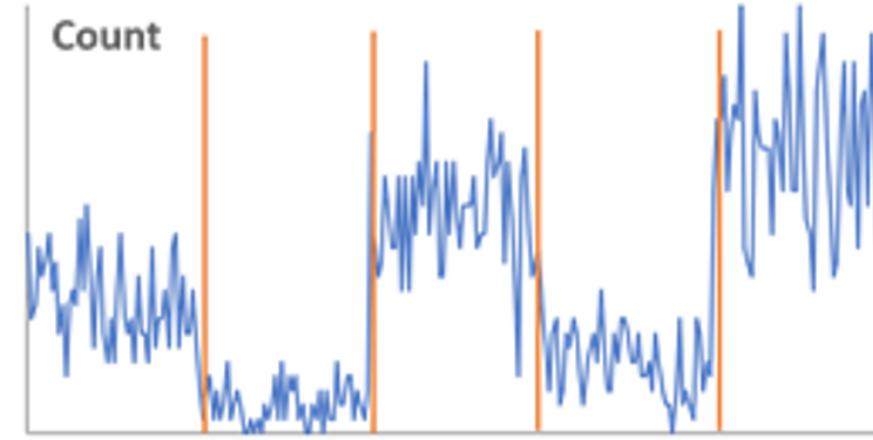
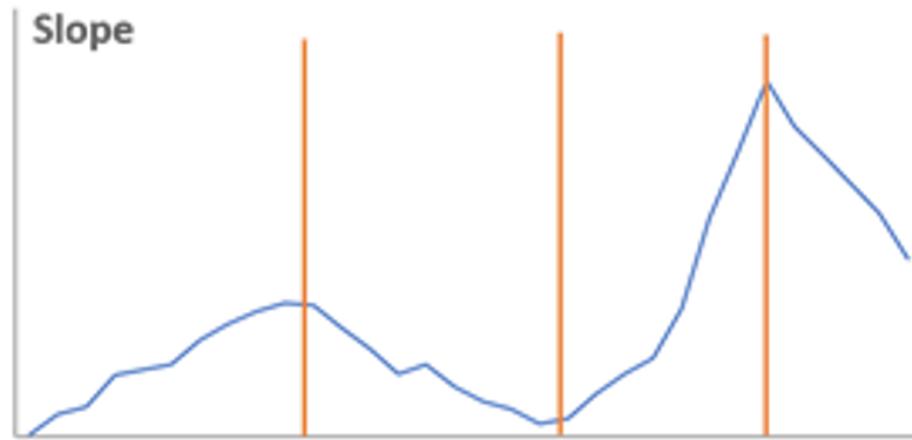
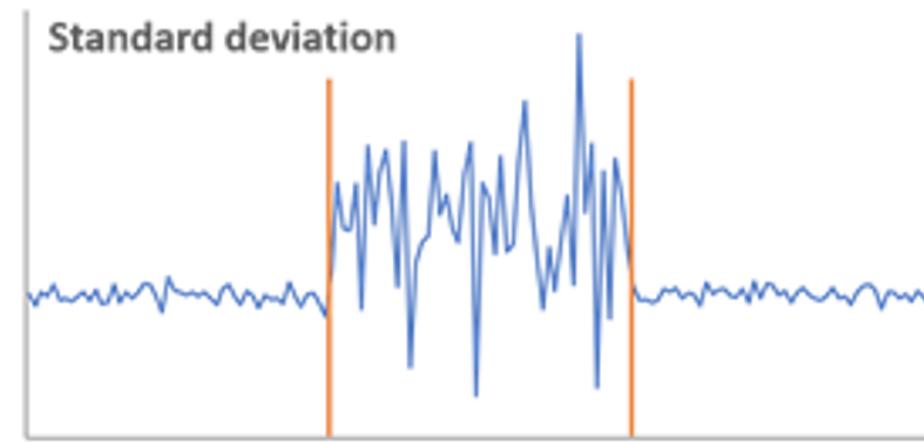
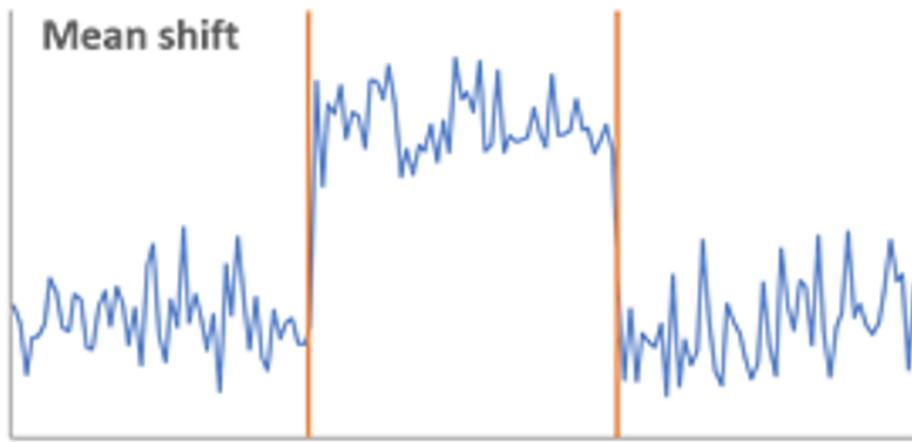
Применяем любые методы для  
кластеризации табличных данных



## Задачи на временных рядах

- Прогнозирование
- Классификация
- Кластеризация
- Сегментация
- Поиск аномалий

# Сегментация



[1] Ruptures package

[2] A Survey of Methods for Time Series Change Point Detection

# Сегментация

Где могут возникать изменения поведения рядов:

- Стоимость акций
  - Изменение стоимости между важными событиями
- Бизнес метрики
  - Перевод продукта из этапа тестирования на маленькой группе на всех клиентов
- Резкая смена настроений в обществе

# Сегментация

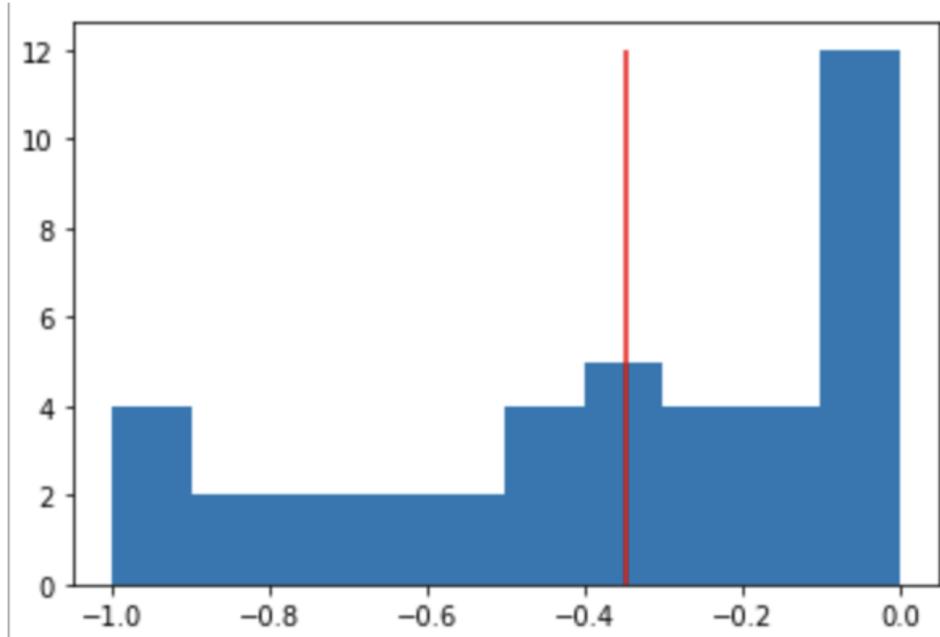
Где может быть полезна:

- Прогнозирования
  - Обрезание ряда для более релевантной истории
- Анализ
- Диаризация спикеров

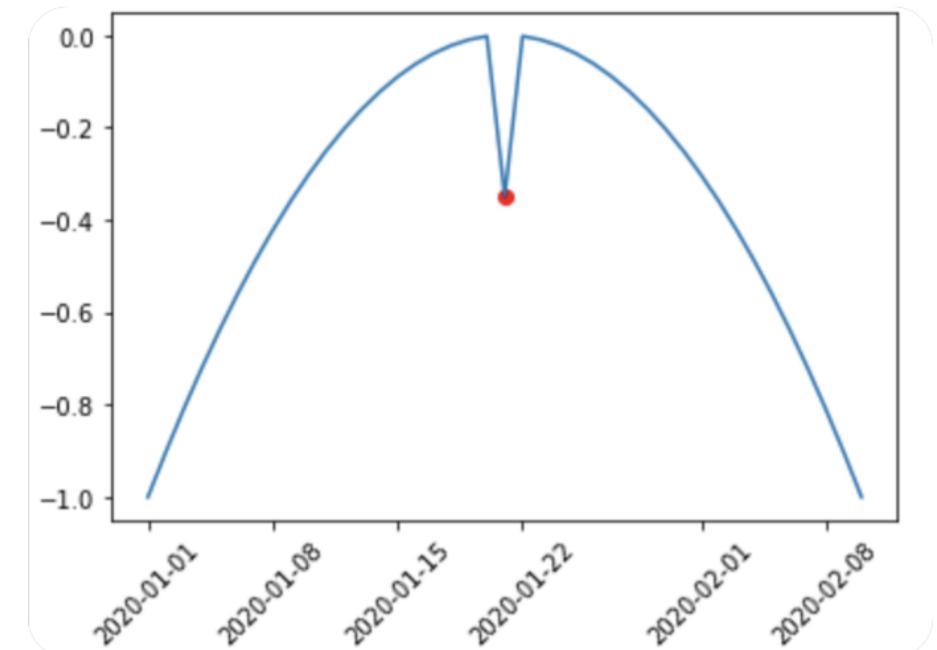
## Задачи на временных рядах

- Прогнозирование
- Классификация
- Кластеризация
- Сегментация
- Поиск аномалий

# Поиск аномалий

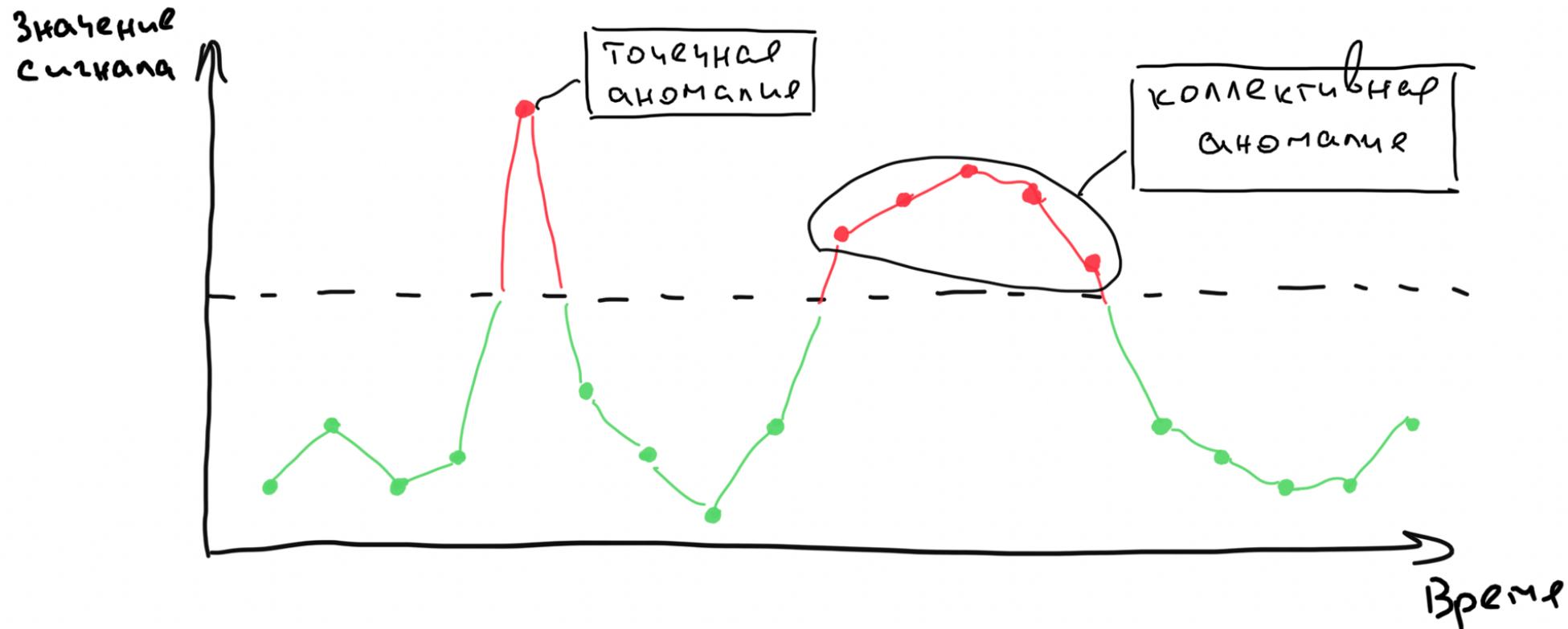


Распределение данных без временной  
метки



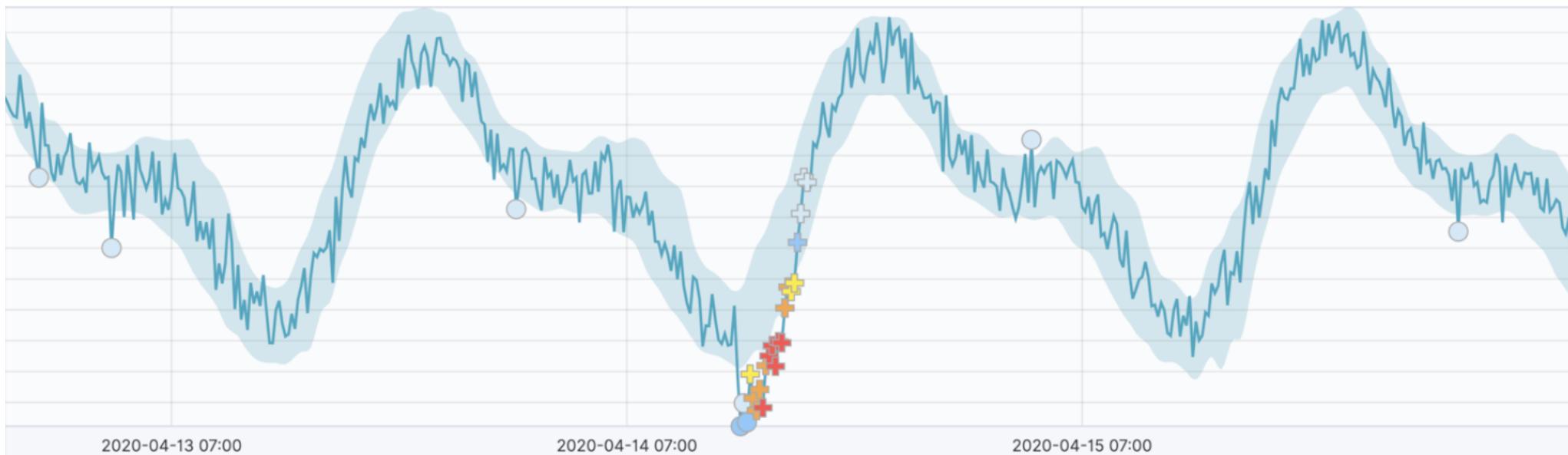
Данные с временной меткой

# Поиск аномалий (оффлайн)

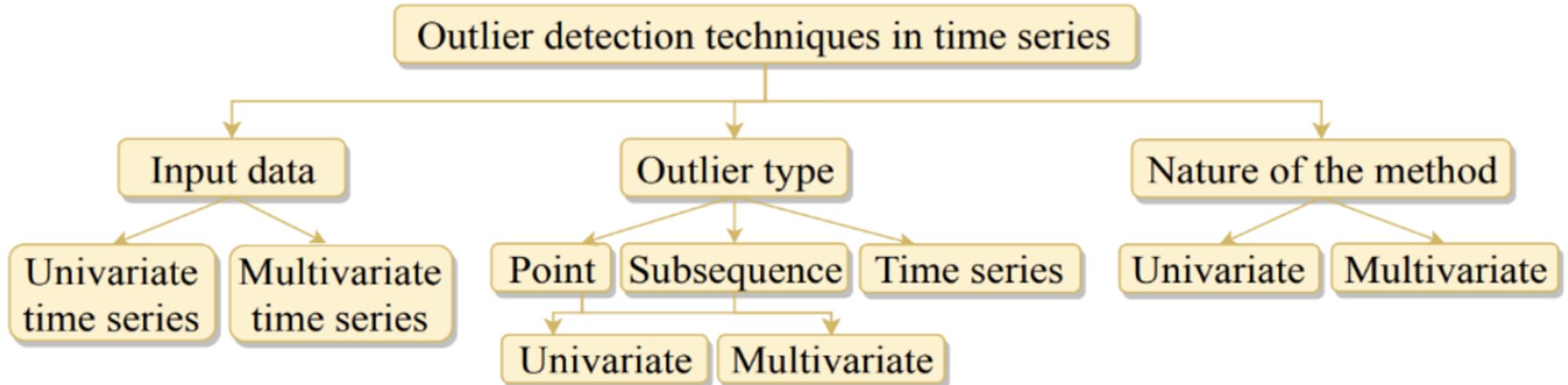


# Поиск аномалий (оффлайн)

Пример коллективной аномалии

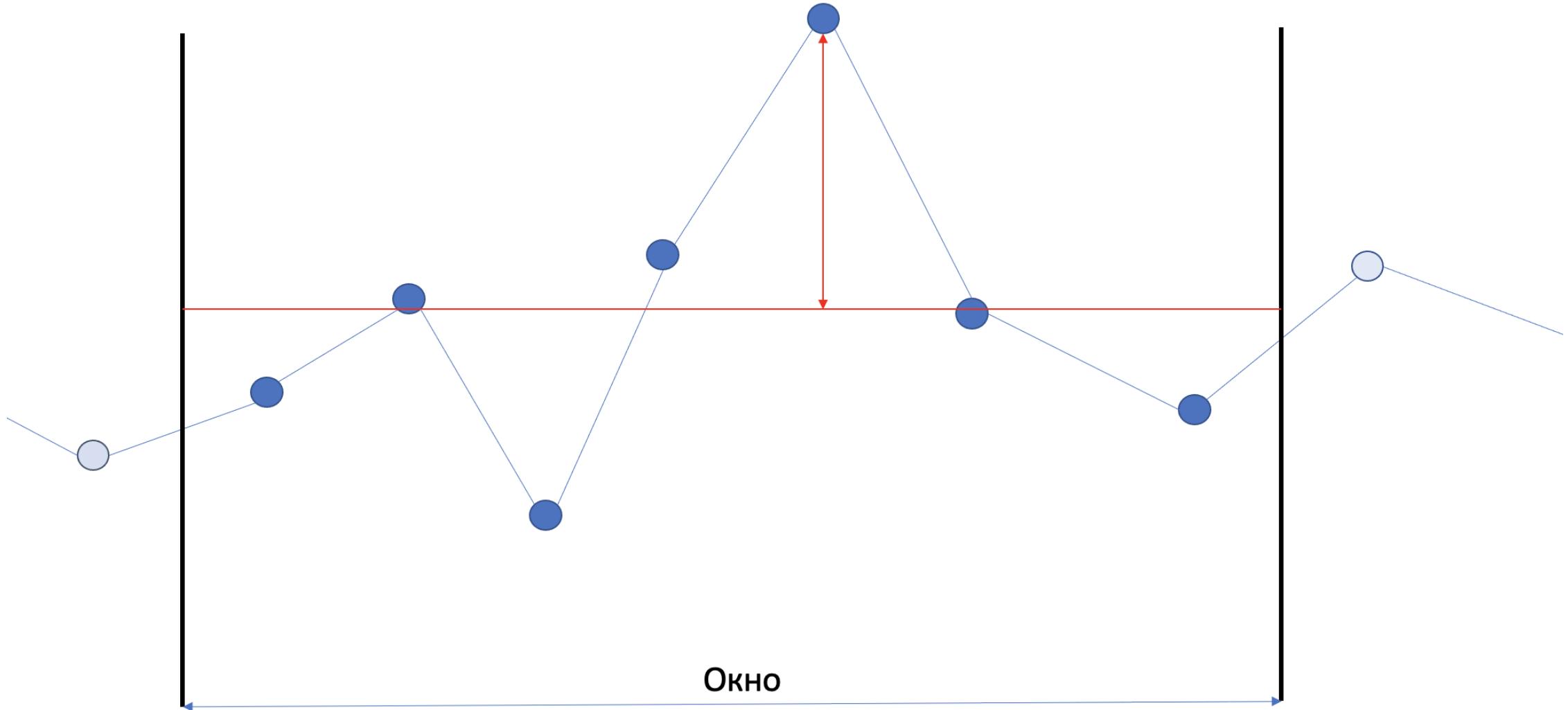


# Поиск аномалий (оффлайн)



[1] A review on outlier/anomaly detection in time series data

# Поиск аномалий (оффлайн)

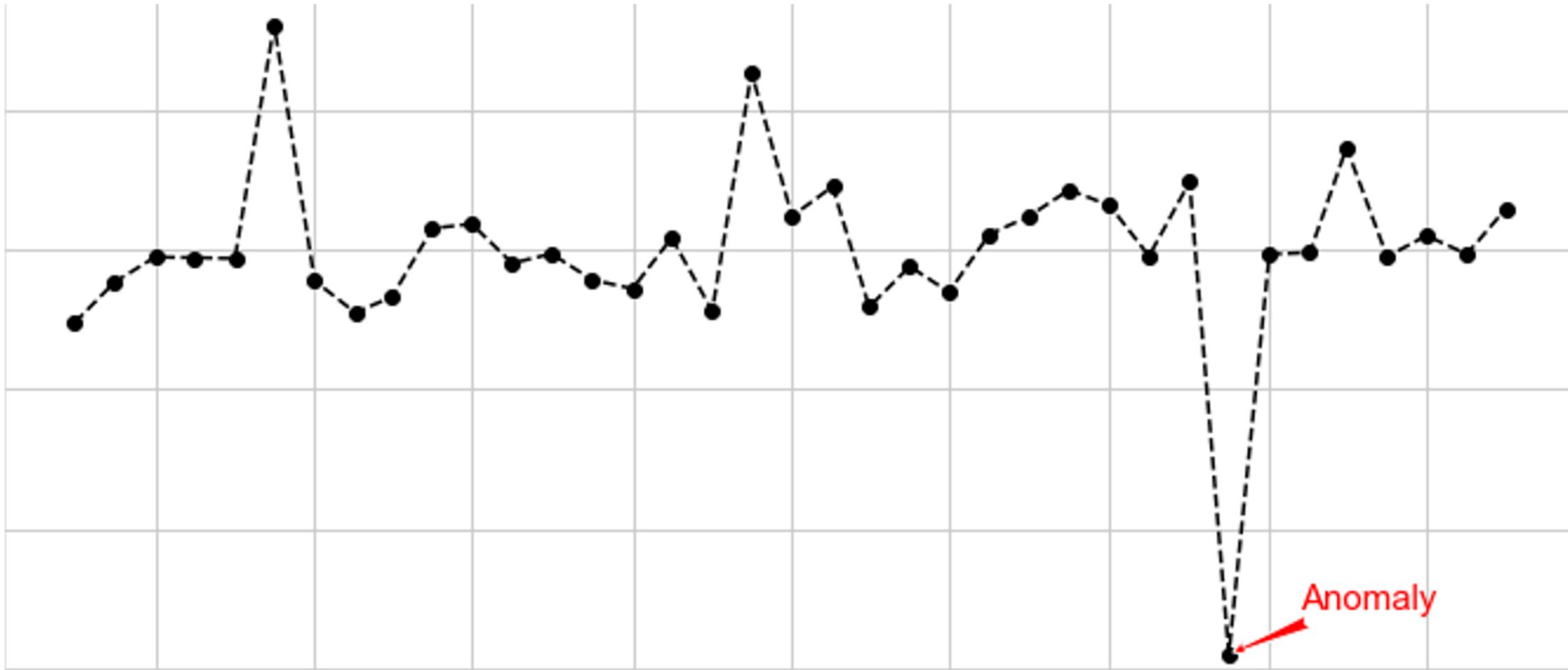


## Поиск аномалий (оффлайн)

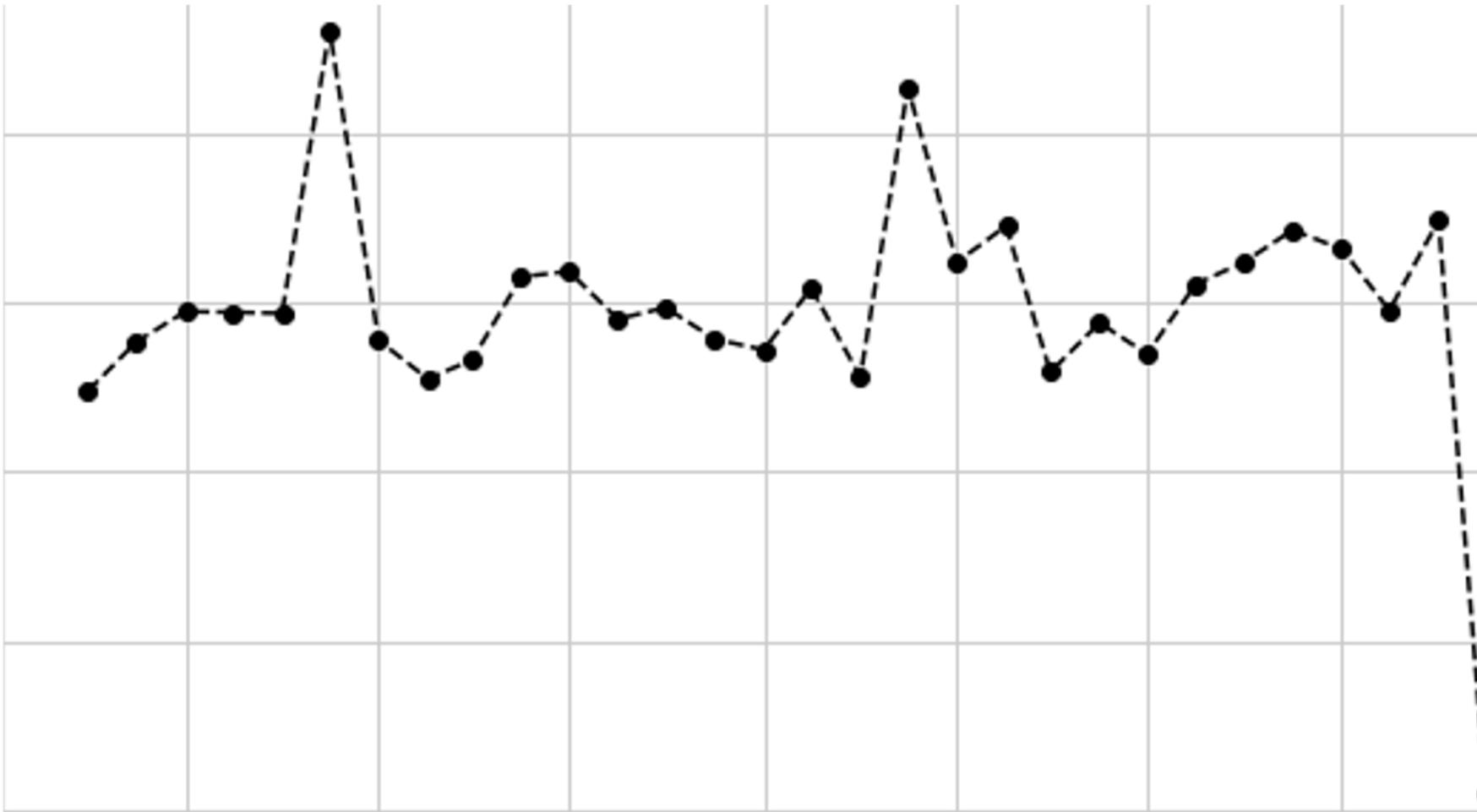
Где может быть полезен:

- Прогнозирования
  - Обработка выбросов для более релевантной истории
- Анализ
  - Поиск аномалий для дальнейшего разбора причин появления

## Поиск аномалий (оффлайн)



## Поиск аномалий (онлайн)



## Поиск аномалий (онлайн)

Где может быть полезен:

- Любой мониторинг

## Задачи на временных рядах

- Прогнозирование
- Классификация
- Кластеризация
- Сегментация
- Поиск аномалий

# Задача прогнозирования временных рядов

## Постановка задачи прогнозирования

$H$  - горизонт прогнозирования

$t_0$  - время старта прогнозирования

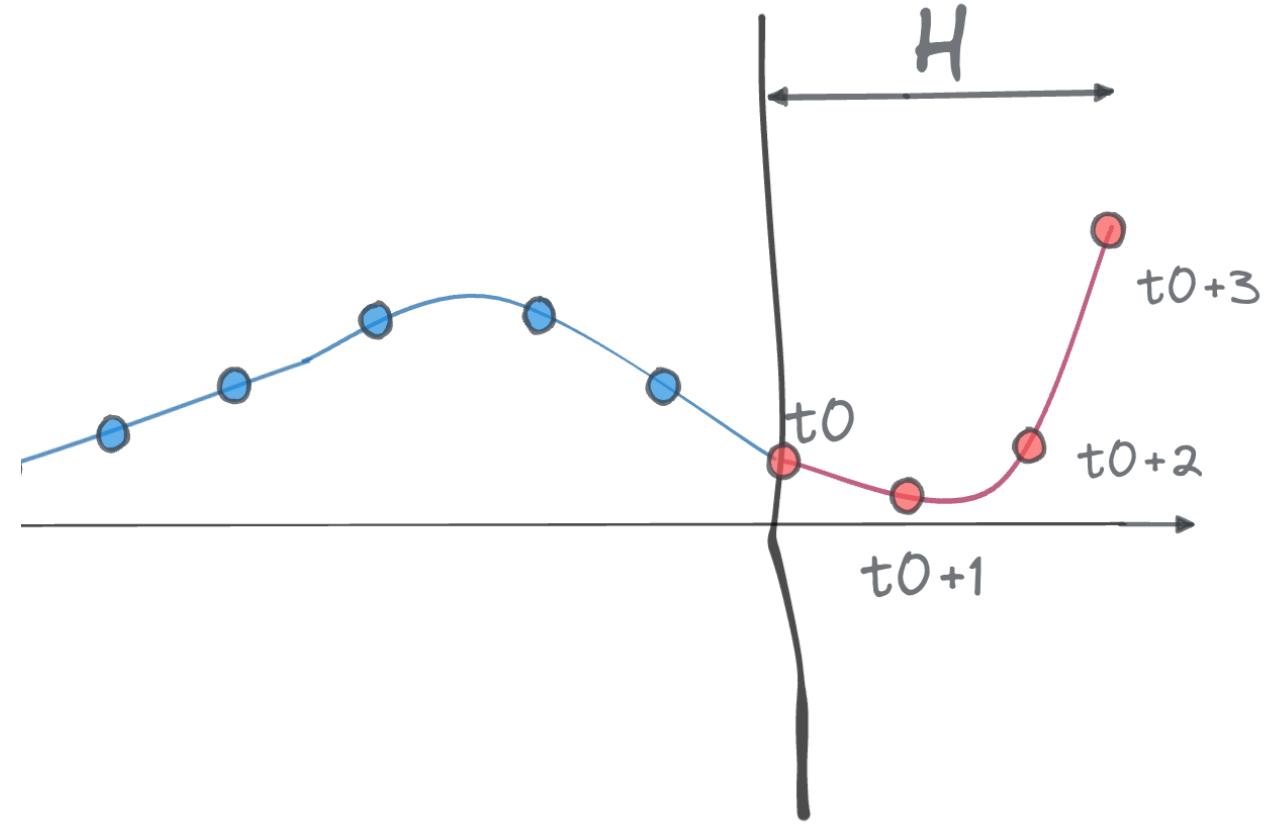
$y_t \in \mathbb{R}^1$  - значение временного

ряда в момент времени  $t$

$\mathbf{x}_t \in \mathbb{R}^n$  - признаки в момент  
времени  $t$

$\hat{y}_{t_0:t_0+H} = f(\mathbf{x}_{0:t_0+H}, y_{0:t_0})$  -

прогноз на горизонт  $H$



## Метрики

$MAE = \frac{1}{T} \sum_{t=1}^T |y_t - \hat{y}_t|$  - средняя абсолютная ошибка

$MSE = \frac{1}{T} \sum_{t=1}^T (y_t - \hat{y}_t)^2$  - средняя квадратичная ошибка

$MAPE = \frac{1}{T} \sum_{t=1}^T \frac{|y_t - \hat{y}_t|}{y_t}$  - процентная ошибка

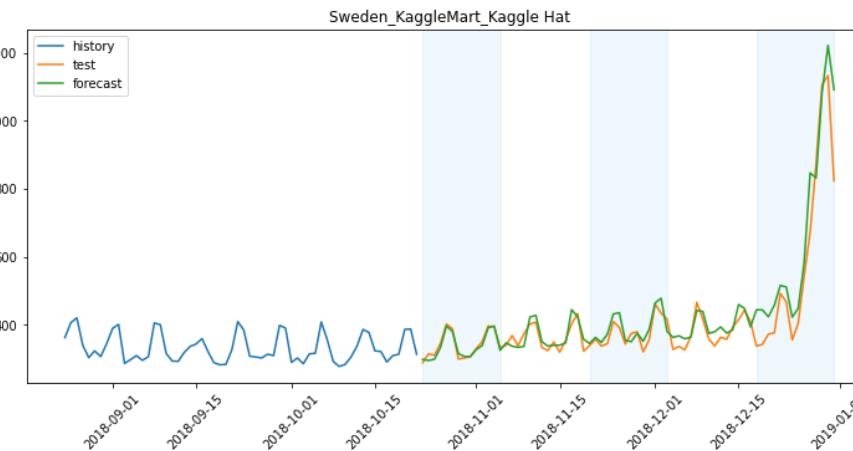
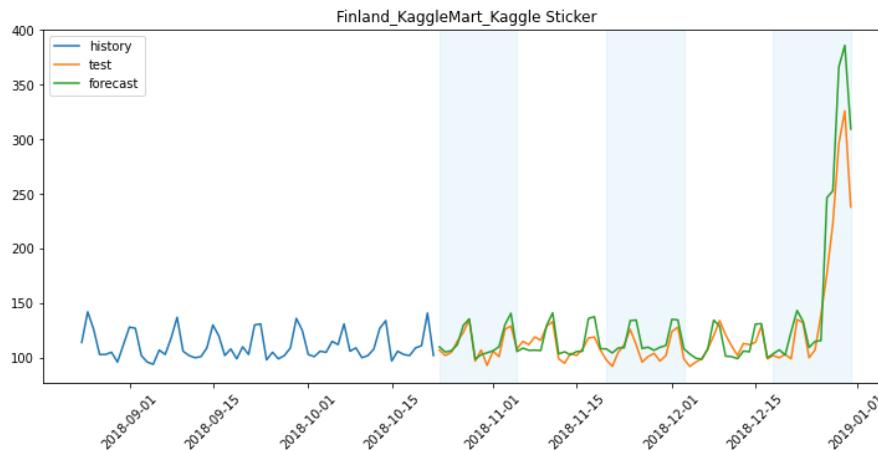
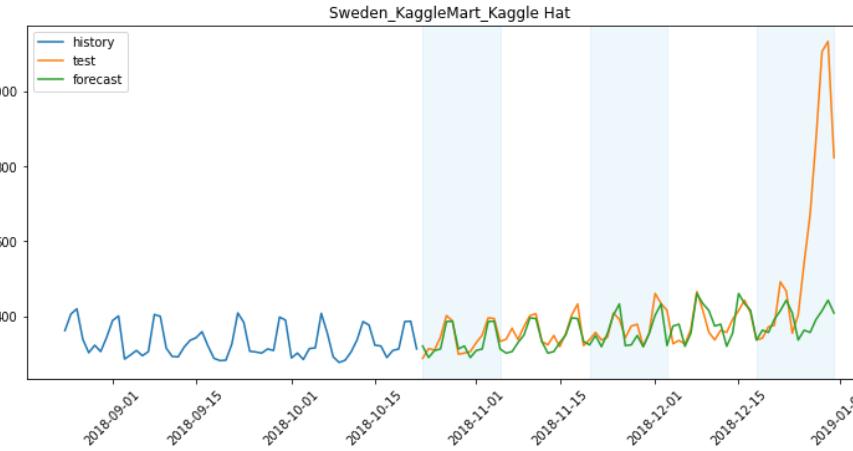
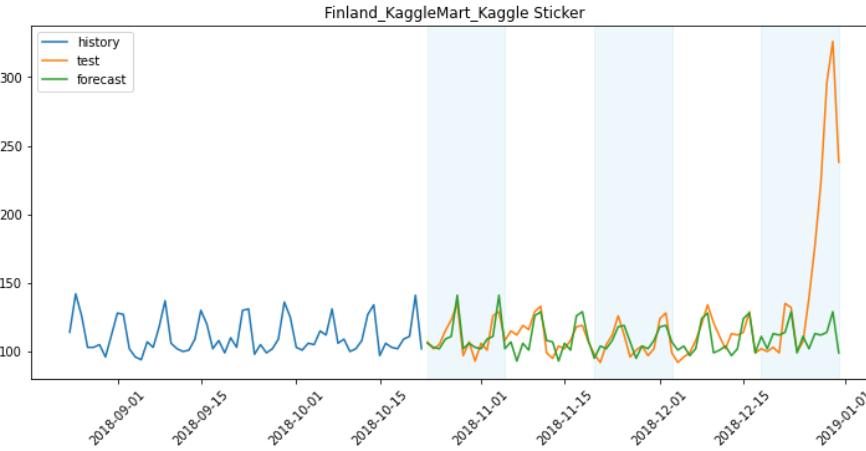
$SMAPE = \frac{2}{T} \sum_{t=1}^T \frac{|y_t - \hat{y}_t|}{|y_t| + |\hat{y}_t|}$  - симметричная процентная ошибка

$ND = MAE / \bar{y}$  - пытаемся учитывать масштаб данных

И менее популярные:  $MSLE$ ,  $MedAE$ ,  $R^2$ ,  $MaxDeviation$  и т.д.

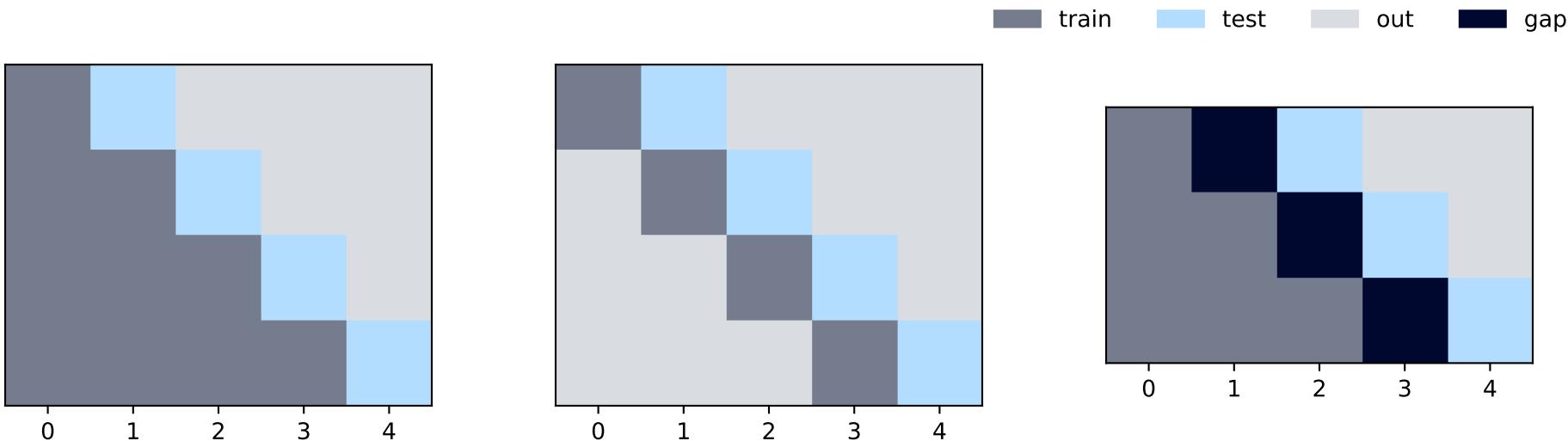
# Пример

## 11.5 vs 6.3 SMAPE



# Валидация

## Как разбивать на фолды?



! Помним, что данные имеют временную структуру и мы не можем перемешивать блоки, чтобы случайно не заглянуть в будущее и не получить leak

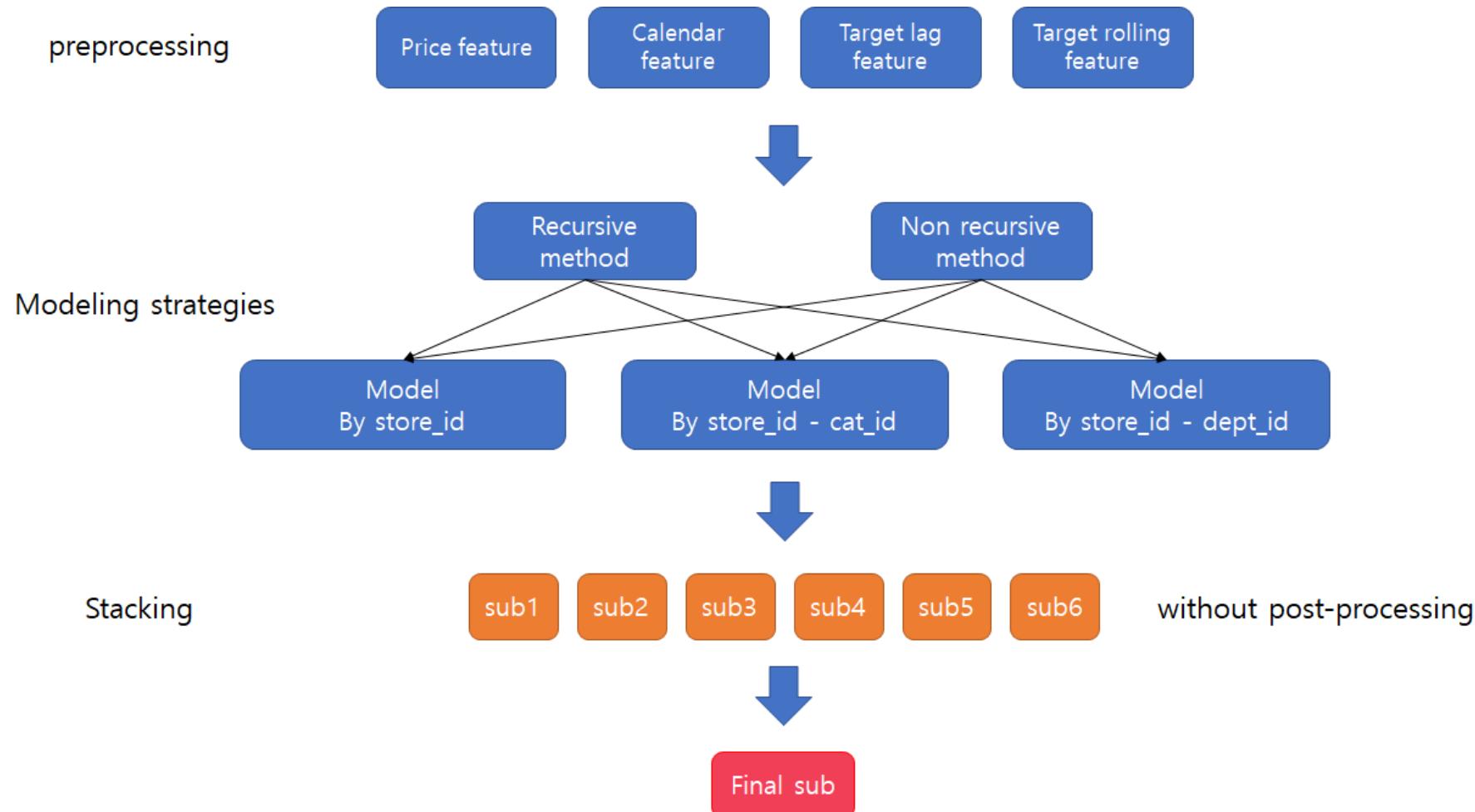
[1] Evaluating time series forecasting models

## Стратегии прогнозирования

- Регрессионная - сводим задачу к задаче табличных данных, дальше используем стандартные методы для задачи регрессии
  - Отдельная модель на каждый шаг прогнозирования
  - Одна модель на все шаги прогнозирования
- Авторегрессионная - предсказываем на шаг вперед, вызываем модель на предсказанных значениях - повторяем так  $H$  раз

# Какой подход выбрать

Пример первого места из соревнования M5



## Подходы прогнозирования: сколько моделей нужно?

- Глобальные модели - одна модель на все данные
- Локальные модели - одна модель на каждый временной ряд

## Достоинства и недостатки глобальных моделей

- Достоинства
  - Быстрее обучаются
  - Удобно использовать в production
  - Тolerантны к длине рядов и к их качеству
  - Возможный перенос знаний между рядами
  - Можно использовать для прогнозирования ряда, который не был в обучающей выборке
- Недостатки
  - Большая вероятность недообучения
  - Ряды могут быть сильно разными по своей природе - из разных доменов -  
нужен дополнительный анализ

## Достоинства и недостатки локальных моделей

- Достоинства
  - Можно хорошо обучиться под конкретный домен или ряд [1]
  - Понятно как параллельно обучать
- Недостатки
  - Медленнее обучаются
  - Сложно использовать в production, если рядов становится много
  - Большее требований к количеству данных и качеству данных

| [1] MixSeq: Connecting Macroscopic Time Series Forecasting with Microscopic Time Series Data

## Пример M4

Постановка задачи: построить прогноз для 100000 рядов разной частотности и разных доменов

- Первое место - Hybrid ES-RNN - глобальная модель, обученная на всех рядах сразу [1]
- Второе место - ансамбль локальных моделей [2]

[1] [Introducing a New Hybrid ES-RNN Model](#)

[2] [The M4 Competition: Results, findings, conclusion and way forward](#)

[3] [Combination of Forecast Methods by Feature-based Learning](#)

## Методы прогнозирования временных рядов

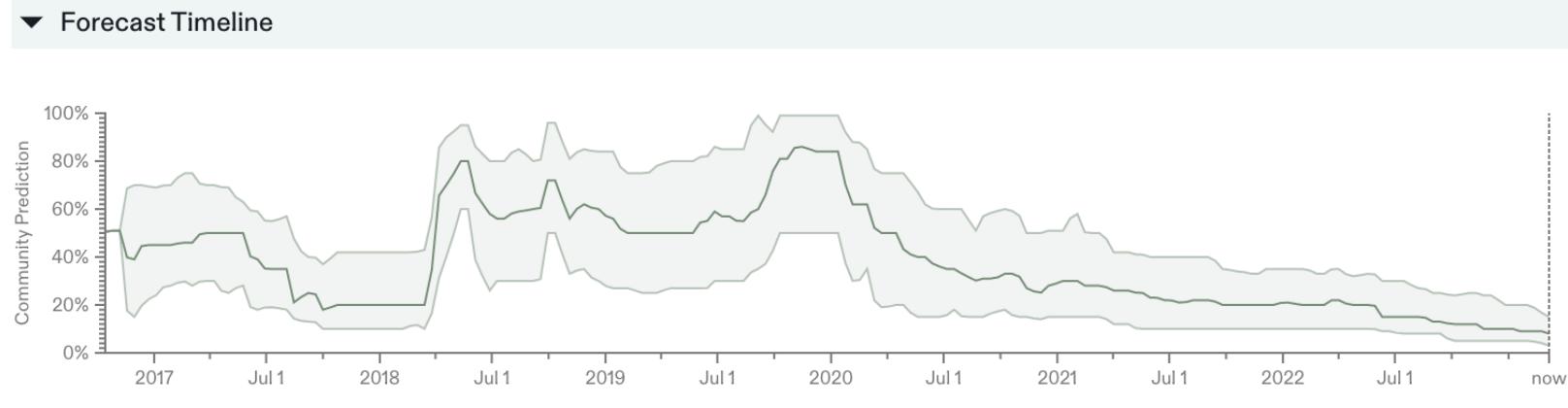


# Предсказания толпы: Wisdom of the Crowd

## SpaceX Lands People on Mars by 2030

Created by Greg • Opened Oct 13, 2016 • Resolves Jan 2, 2030

^ 884 ▼ Open Closes Jan 1, 2030



Total Predictions

6.24k

Total Forecasters

3795

Community Prediction 8%

Community

- 2018 - запуск Tesla в космом, 2019-2020 - пилотируемые запуски к МКС и успешные испытания Starship в 2020

[1] [SpaceX Lands People on Mars by 2030](#)

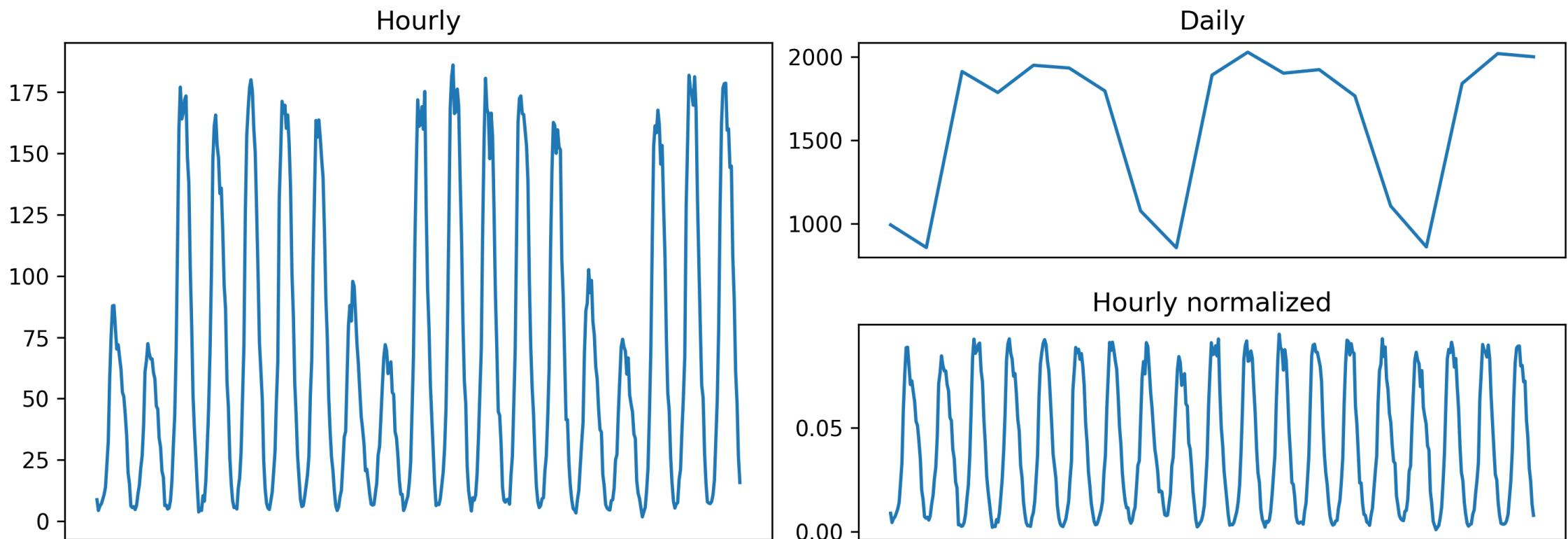
## Rule based или heuristic based подходы

- Данных очень мало
- Эксперты и их знания, которые сложно переложить на модели - такое часто встречается в анализе спроса или финансовом моделировании
- Простые процессы, которые легко формализовать через `if else` правила

```
if timestamp in night:  
    return 0  
elif timestamp in morning:  
    return 5  
elif timestamp in day:  
    return 20  
elif timestamp in evening:  
    return 10  
else:  
    raise ValueError("Unknown timestamp")
```

## Моделирование распределений

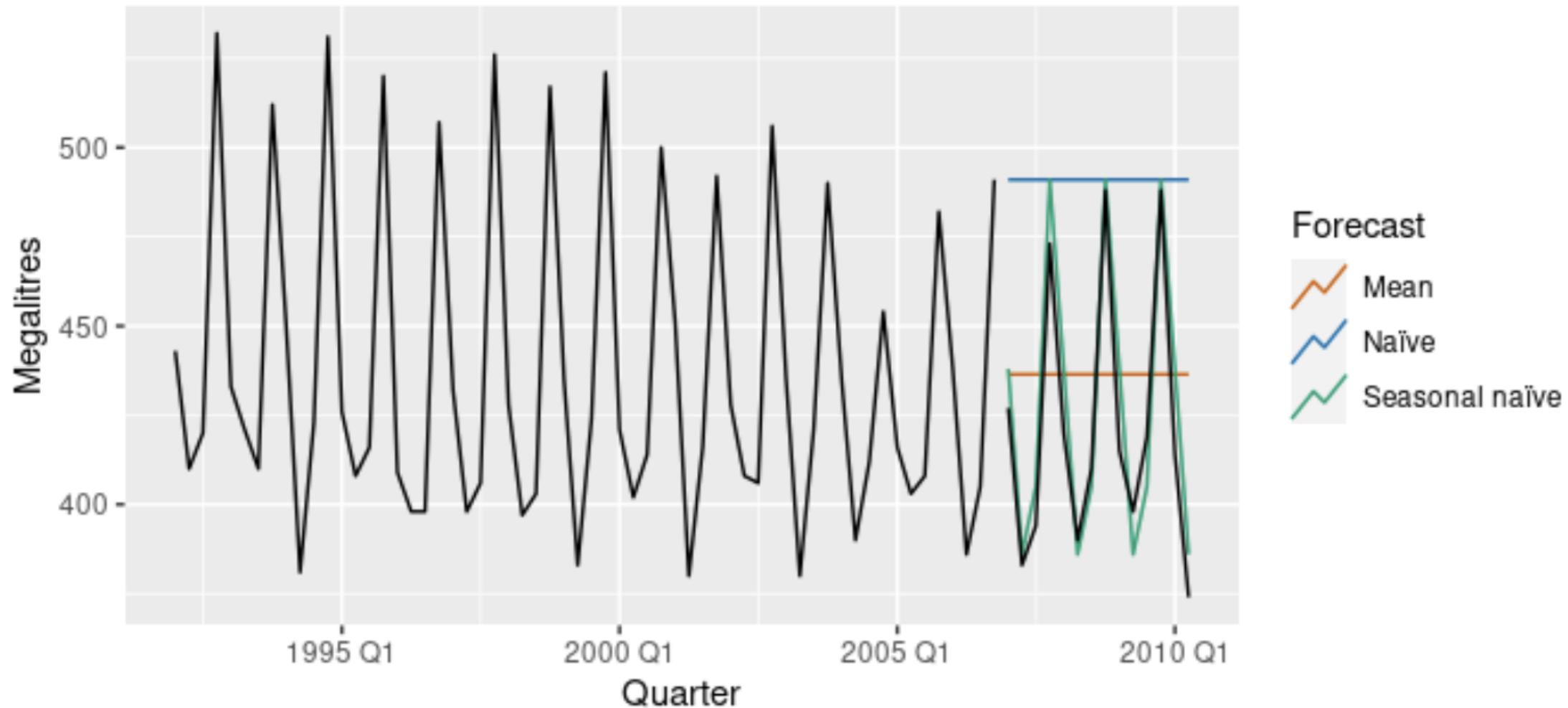
Пример: звонки в коллцентр и распределение в течение дня - для прогнозирования доли звонков в течение дня используем распределение Дирихле



## Лаги и скользящие статистики

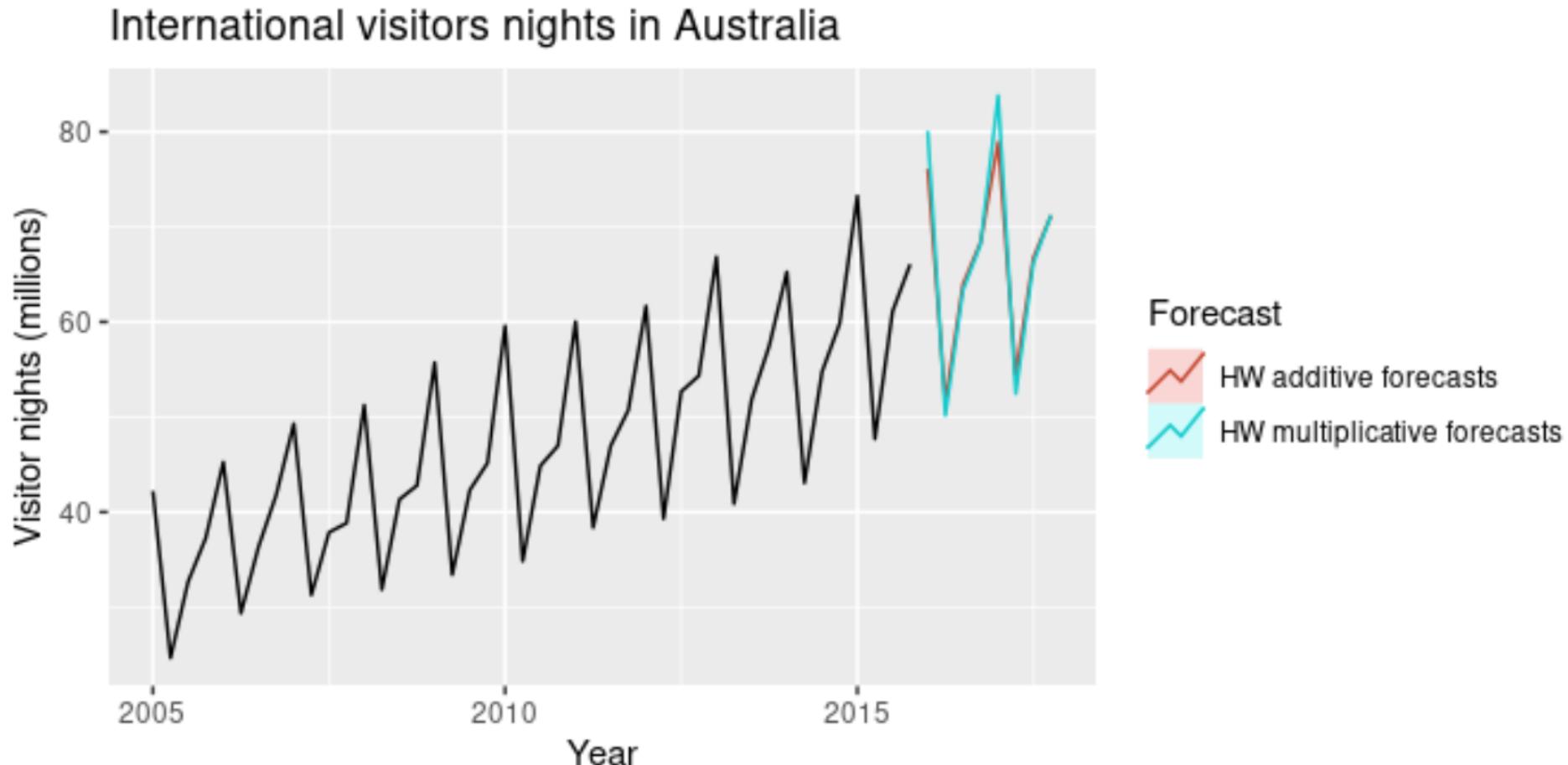
- Хороший безлайн, который не всегда легко побить

Forecasts for quarterly beer production



# Holt-Winters

- Расширение метода экспоненциального сглаживания для временных рядов с добавлением тренда и сезонности
  - Относительно простая модель - три параметра для оптимизации



# ETS и HW

Trend		Seasonal	
	N	A	M
	$\hat{y}_{t+h t} = \ell_t$	$\hat{y}_{t+h t} = \ell_t + s_{t+h-m(k+1)}$	$\hat{y}_{t+h t} = \ell_t s_{t+h-m(k+1)}$
N	$\ell_t = \alpha y_t + (1 - \alpha) \ell_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1}) + (1 - \gamma)s_{t-m}$	$\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)\ell_{t-1}$ $s_t = \gamma(y_t/\ell_{t-1}) + (1 - \gamma)s_{t-m}$	$\ell_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)\ell_{t-1}$ $s_t = \gamma(y_t/\ell_{t-1}) + (1 - \gamma)s_{t-m}$
	$\hat{y}_{t+h t} = \ell_t + hb_t$	$\hat{y}_{t+h t} = \ell_t + hb_t + s_{t+h-m(k+1)}$	$\hat{y}_{t+h t} = (\ell_t + hb_t)s_{t+h-m(k+1)}$
A	$\ell_t = \alpha y_t + (1 - \alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1}$	$\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1}$	$\ell_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)b_{t-1}$
	$\hat{y}_{t+h t} = \ell_t + \phi_h b_t$	$\hat{y}_{t+h t} = \ell_t + \phi_h b_t + s_{t+h-m(k+1)}$	$\hat{y}_{t+h t} = (\ell_t + \phi_h b_t)s_{t+h-m(k+1)}$
<b>A<sub>d</sub></b>	$\ell_t = \alpha y_t + (1 - \alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)\phi b_{t-1}$	$\ell_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)\phi b_{t-1}$	$\ell_t = \alpha(y_t/s_{t-m}) + (1 - \alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1 - \beta^*)\phi b_{t-1}$
		$s_t = \gamma(y_t - \ell_{t-1} - \phi b_{t-1}) + (1 - \gamma)s_{t-m}$	$s_t = \gamma(y_t/(\ell_{t-1} + \phi b_{t-1})) + (1 - \gamma)s_{t-m}$

## ARIMA

$$Y_t = (1 - L)^d X_t \quad LX_t = X_{t-1} \text{ -- differencing operator}$$

$$\left(1 - \sum_{i=1}^p \varphi_i L^i\right) Y_t = \left(1 + \sum_{i=1}^q \theta_i L^i\right) \varepsilon_t, \quad (\varphi_i, \theta_j)$$

- Включает в себя семейство ETS моделей - моделей с экспоненциальным сглаживанием и Holt-Winters - мультипликативная версия не включается
- Умеет работать с сезонностью и внешними признаками
- Настройка параметров - как правило используется перебор по сетке с использованием информационного критерия Акаике -  
 $AIC = 2 \#params - 2 \ln(L)$ , а не кросс-валидация
- Плохо работает с выбросами и нестационарными рядами - нужно дополнительно предобрабатывать данные - логарифмировать, обрезать историю и т.д.

## Prophet

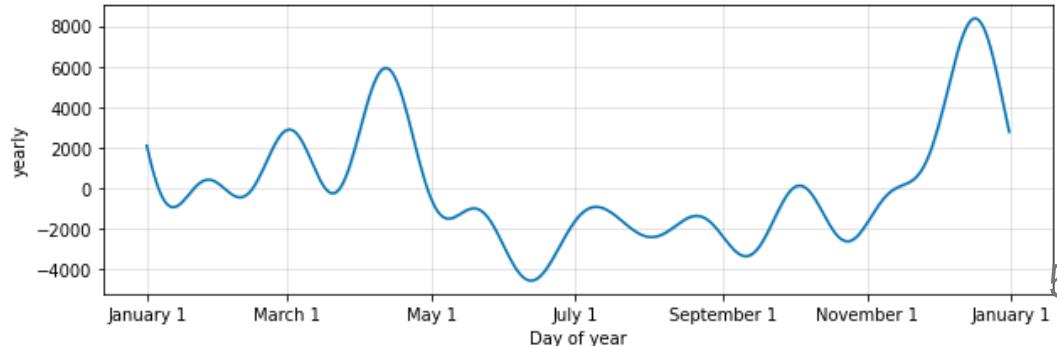
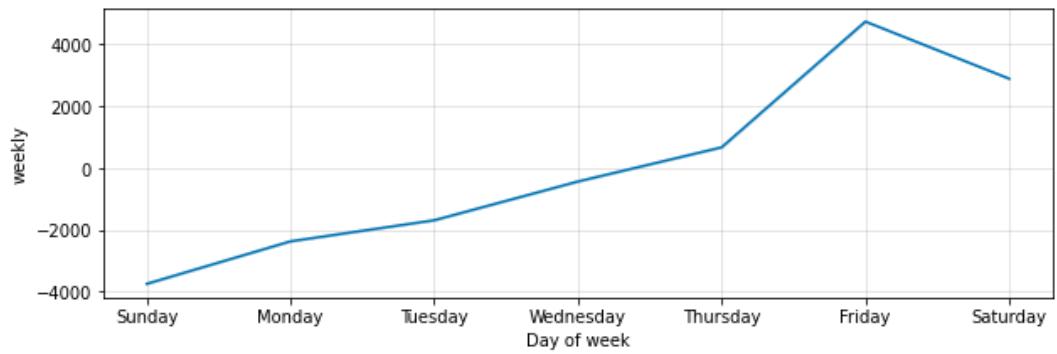
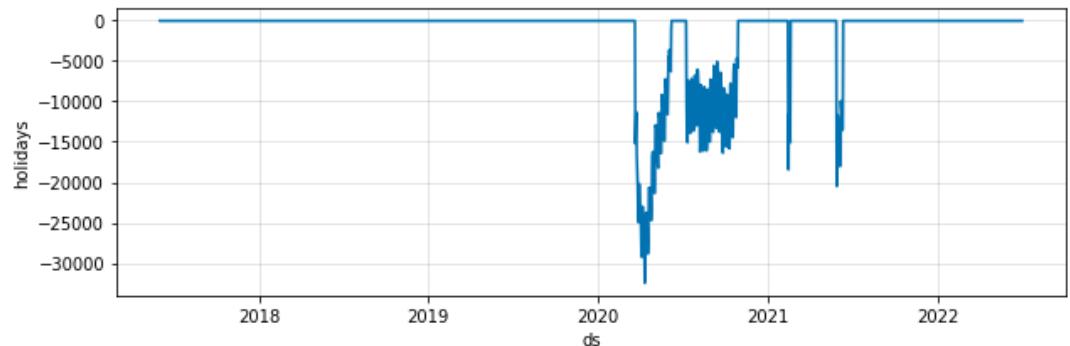
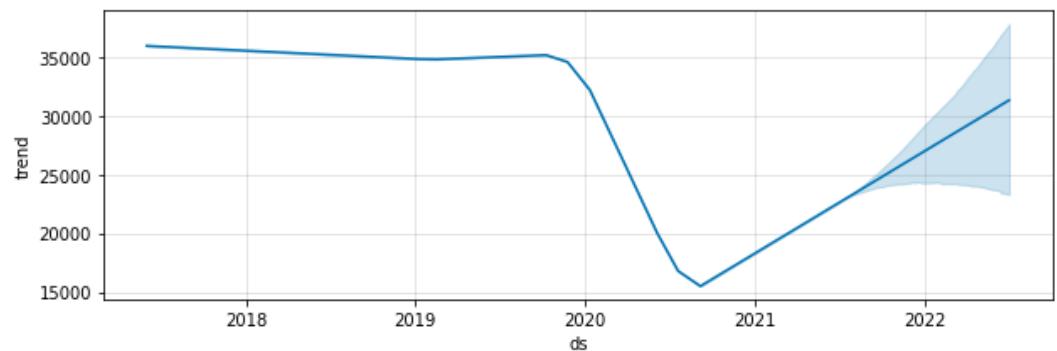
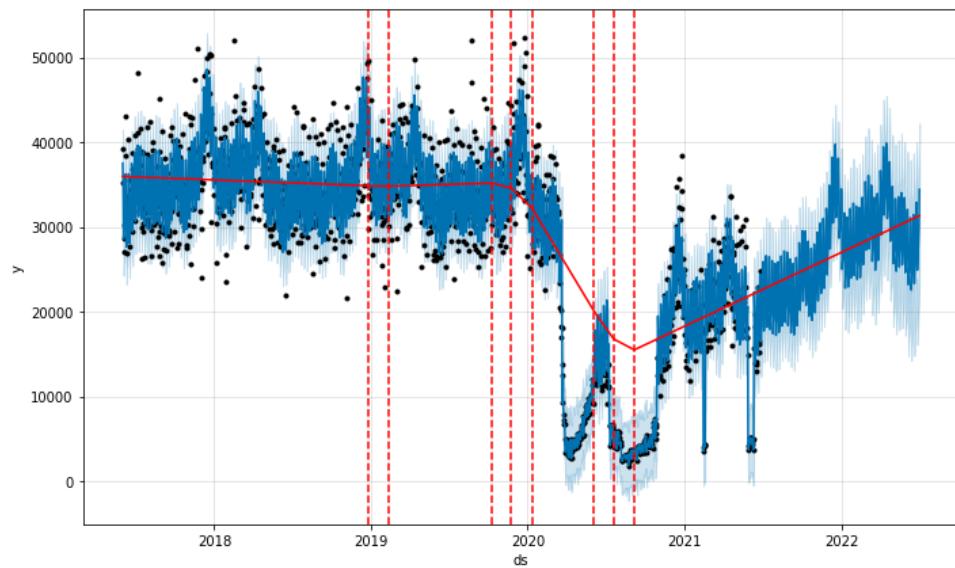
- Понятная и хорошо интерпретируемая модель
- Умеет работать с сезонностью и внешними признаками
- Умеет работать с выбросами, нестационарными рядами и точками смены тренда
- В основной библиотеке `prophet` на `python` хороший набор инструментов для анализа прогнозов
- Возможность ручной настройки поведения модели в точках смены тренда, сезонности и праздников

[1] [Prophet: Forecasting at scale](#)

# Prophet

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

- $g(t)$  - тренд
- $s(t)$  - сезонность
- $h(t)$  - праздники



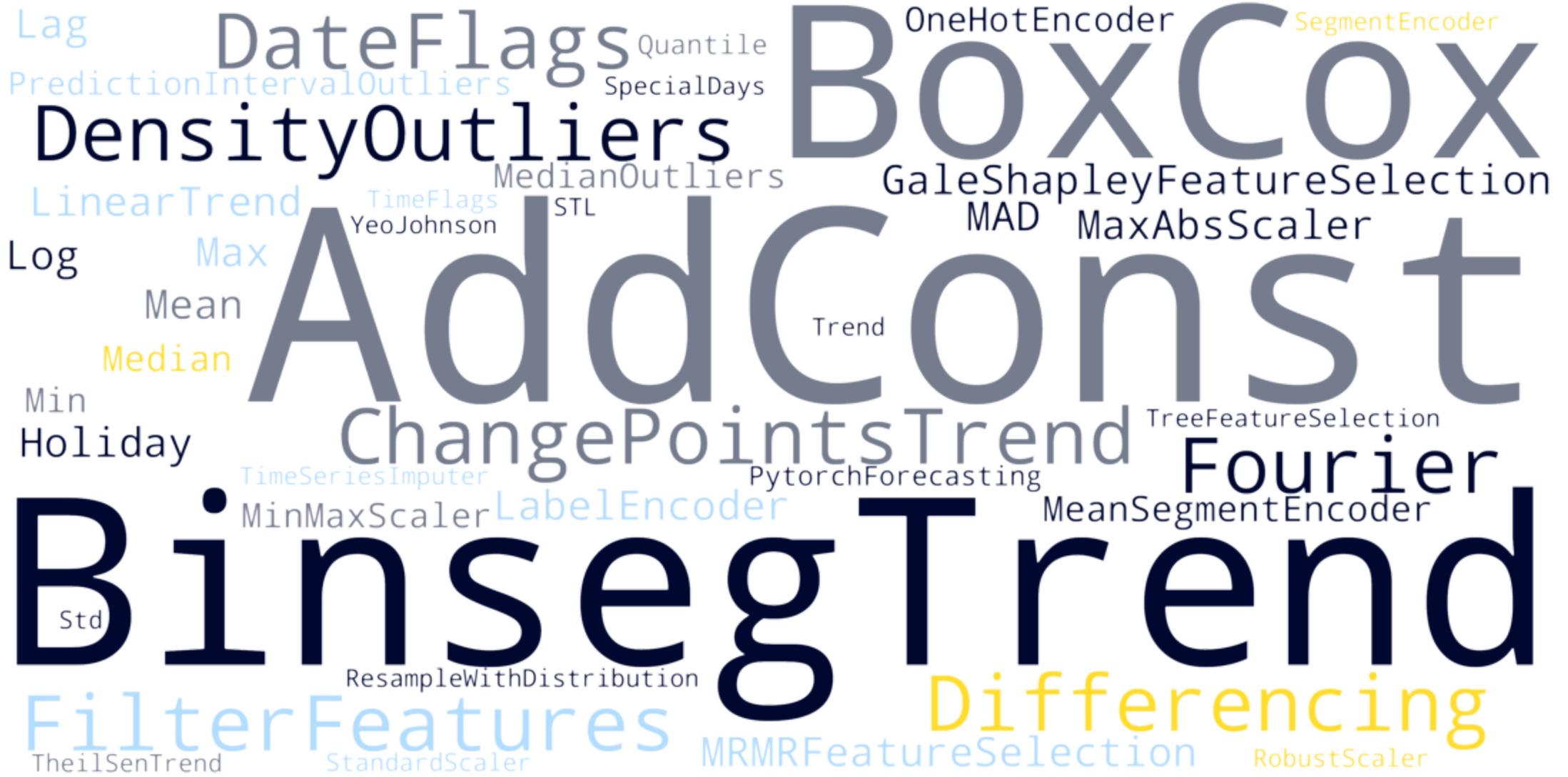
## Задача прогнозирования временные ряды как табличные данные

Идея: почему бы не применить к временным рядам те же методы, что и к табличным данным, с которыми мы умеем работать? Какие признаки можно построить?

Природа признаков:

- Признаки, полученные из самого временного ряда
- Признаки, полученные из дополнительных данных

Out[4]:	segment				Finland_KaggleMart_Kaggle Hat				Finland_KaggleMart_Kaggle Mug			
	feature	precipitation	snow_depth	target	tavg	tmax	tmin	precipitation	snow_depth	target	tavg	.
timestamp												.
2015-01-01	1.714141	284.545455	520.0	1.428571	2.912739	-1.015287	1.714141	284.545455	329.0	1.428571	.	
2015-01-02	10.016667	195.000000	493.0	0.553571	2.358599	-0.998718	10.016667	195.000000	318.0	0.553571	.	
2015-01-03	3.956061	284.294118	535.0	-1.739286	0.820382	-3.463871	3.956061	284.294118	360.0	-1.739286	.	
2015-01-04	0.246193	260.772727	544.0	-7.035714	-3.110828	-9.502581	0.246193	260.772727	332.0	-7.035714	.	
2015-01-05	0.036364	236.900000	378.0	-17.164286	-8.727564	-19.004487	0.036364	236.900000	243.0	-17.164286	.	



[1] Etna library feature generators and preprocessings

## Какие признаки можно построить из временного ряда

- Лаги
- Признаки временной метки - номер дня, недели, месяца, года и т.д.
- Бегущие статистики - среднее, минимум, максимум, стандартное отклонение, дисперсия, квантили и т.д. - с окном и без окна
- Фурье преобразование - коэффициенты разложения ряда
- Точки смены тренда
- Тренд и сезонные компоненты в качестве признаков

## Какие преобразования можно применить к временному ряду

- Стандартизация
- Удаление тренда - с помощью линейной регрессии
- Удаление сезонности - с помощью stl декомпозиции
- Логарифмирование и Box-Cox для положительных рядов. Yeo-Johnson для рядов в области действительных чисел
- Дифференцирование
- Сглаживание - с помощью скользящего среднего, экспоненциального сглаживания и т.д.

## Лики и откуда они возникают

- Некорректно разделены данные на train и test - например, использовали случайное разбиение - увидели будущее на этапе обучения
- Препроцессинг данных до разделения на train и test - удаление тренда или стандартизация
- Наличие признаков, которые не доступны во время теста - важно разделять признаки и дополнительные данные на известные и неизвестные в будущем

Почему это неприятно - признаки будут не доступны в production среде и модель не будет работать или будет работать некорректно

## Как понять, что что-то пошло не так и как не попасть в ловушку

- В самом начале разделить данные на train и test
- Проверить, что в обучающей и тестовой выборке, после построения признаков, есть все временные метки и что в датасете нет `NaN` там где их не должно быть (например, такое часто случается с лагами)
- Проверить на тестовых данных метрики адекватные - если они лучше чем на обучающих данных, то что-то возможно пошло не так

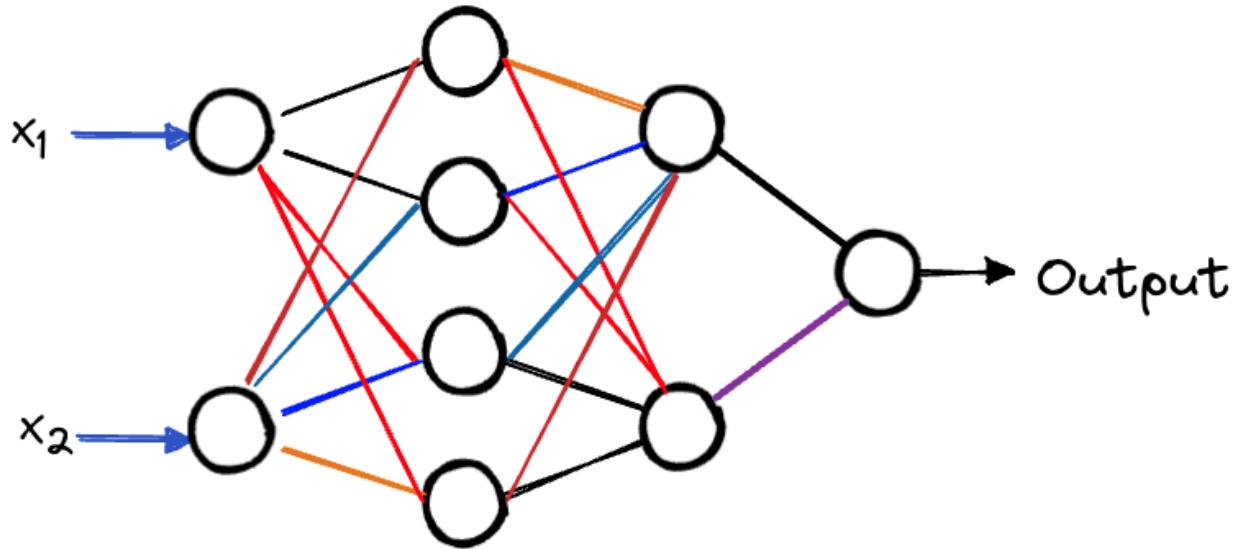
**Главное:** использовать готовые библиотеки для работы с временными рядами - как правило - они уже покрыты тестами и проверены другими пользователями и разработчиками - багов должно быть меньше

Немного про глубокие модели



## MLP модели

Вспоминаем про табличные данные - нужно строить признаки =(



Почему вам могут понадобиться MLP модели:

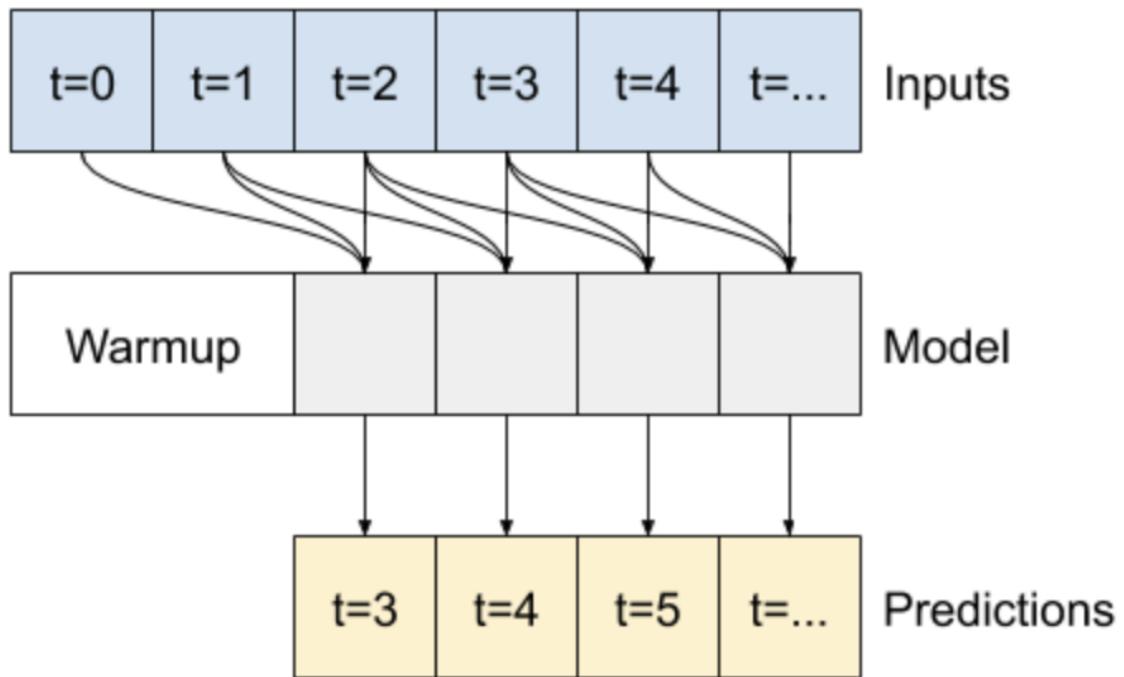
- Проще интегрировать свою функцию потерь, чем в деревья или градиентный бустинг - иногда это может быть важно
- Легко переделать в вероятностную модель, поменяв функцию потерь и последний слой

## Моделирование последовательностей

Мы же занимаемся глубоким  
обучением - зачем нам строить  
признаки руками - модель все сама  
извлечет из данных



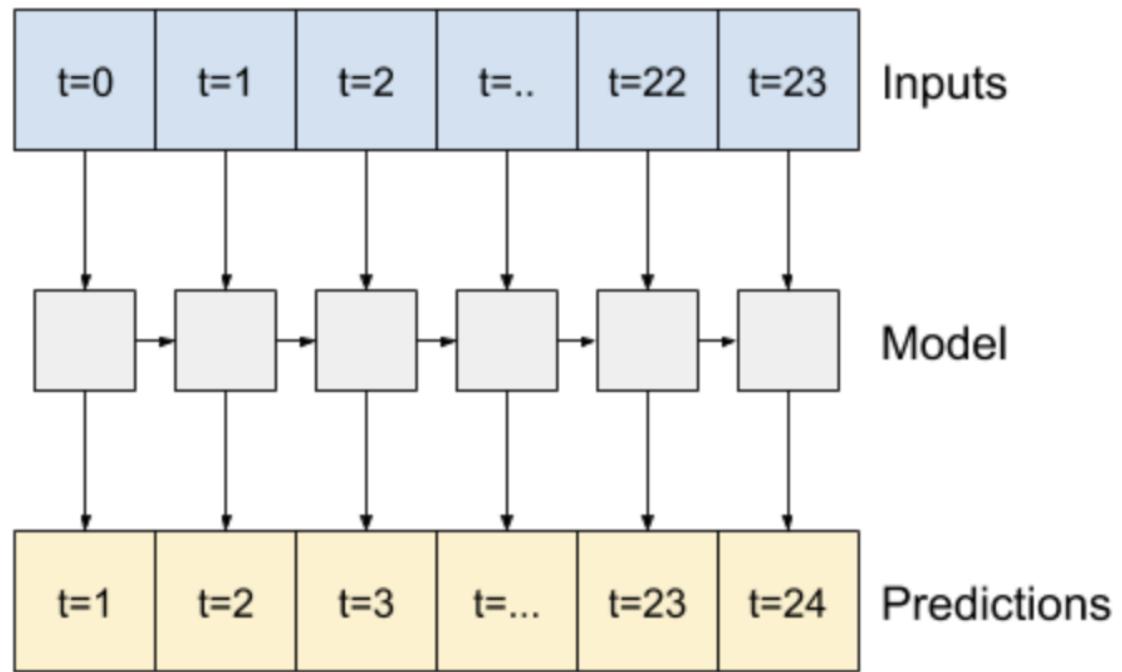
## CNN based модели



- Это расширение классических подходов со скользящим окном
  - но теперь параметры обучаемые и мы можем использовать несколько слоев для увеличение контекста (receptive field)
- В целом - можно пробовать любые архитектуры из speech recognition - например - WaveNet

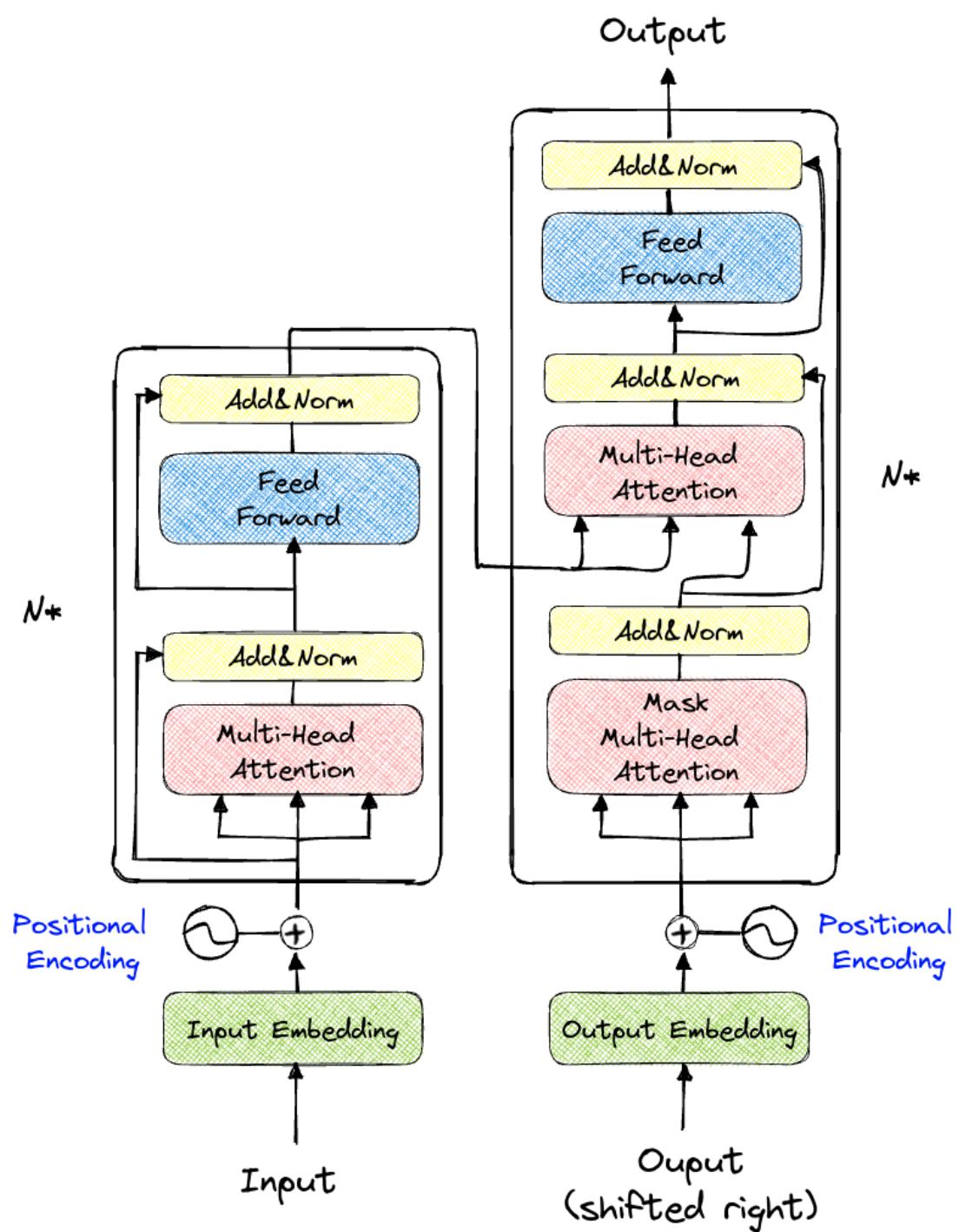
[1] [WaveNet: A Generative Model for Raw Audio](#)

## RNN based модели



- Архитектуры на основе рекуррентных блоков - RNN, GRU, LSTM
- В основном используется для предсказания на шаг вперед - в авторегрессионном режиме
- Долго считалась одним из основных подходов для предсказания временных рядов
- Можно посмотреть на DeepAR и ES-RNN

[1] DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks



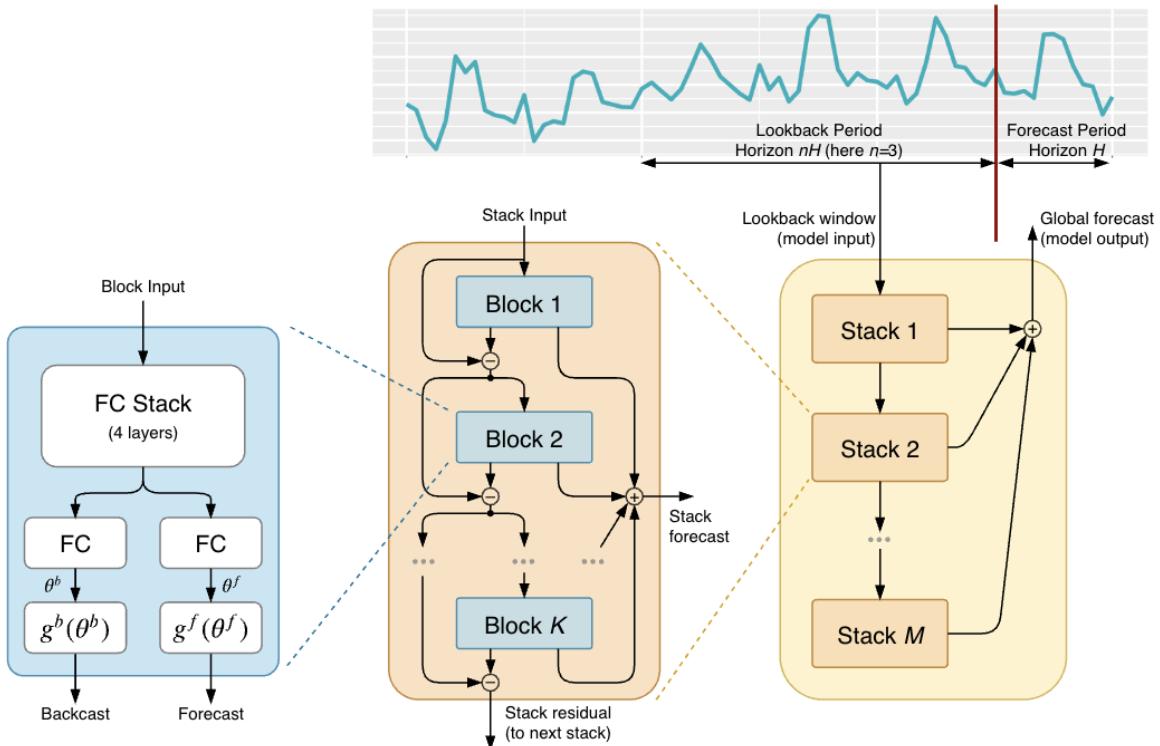
## Transformer based модели

- Позволяет использовать большой контекст
- В основном используется для предсказания на шаг вперед - в авторегрессионном режиме - но можно и предсказывать вектор сразу
- Attention можно использовать для интерпретации модели
- Пример: Temporal Fusion Transformer

[1] Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting

## NBeats

- Модель изначально для предсказания временных рядов
- Активно использует residual connections и стакает много слоев - чем-то похоже на градиентный бустинг
- Показала неплохие результаты на zero-shot предсказаниях



[1] N-BEATS: Neural basis expansion analysis for interpretable time series forecasting

## Про опыт использования в компании и в соревнованиях

Главное: глубокие модели это не панацея - нужно понимать, что вы делаете и какие у вас есть данные

- Например, если у вас есть данные с гео координатами - можно посмотреть на графовые модели - есть успешный опыт
- Почему глубокие модели модели не всегда работают:
  - Очень много общего с табличками - в табличках до сих пор нет явного фаворита
  - Данных не всегда достаточно - у людей в бизнес процессах зачастую около 100 рядов с историей в пару лет в лучшем случае
  - Ансамблирование моделей - используем робастные и сложные методы вместе

## Список ресурсов

- [Forecasting: theory and practice, Amazon and others](#) - сборник с решением прикладных задач
- [Forecasting: Principles and Practice, R.Hyndman](#) – вводный курс по временным рядам
- [ETNA – Time-Series Library](#)
- DL
  - [DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks](#)
  - [Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting](#)
  - [Probabilistic Transformer For Time Series Analysis](#)
  - [Topological Attention for Time Series Forecasting](#)