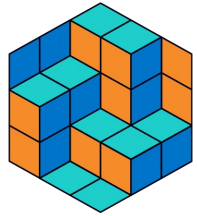
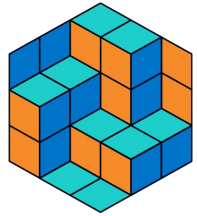


Предсказание временных рядов

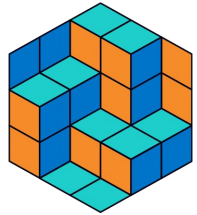
Лекция 3



Глава 1

SARIMA & SARIMAX

ARIMA

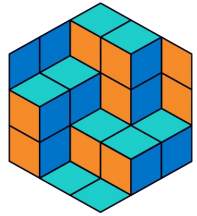


Давайте вспомним кто она такая:

$$\Delta^q Y_t = c + \sum_{i=1}^{p} \Delta^q Y_{t-i} \alpha_i + \sum_{j=0}^{d} b_j \epsilon_{t-j}$$

Помните как она работает?

ARIMA

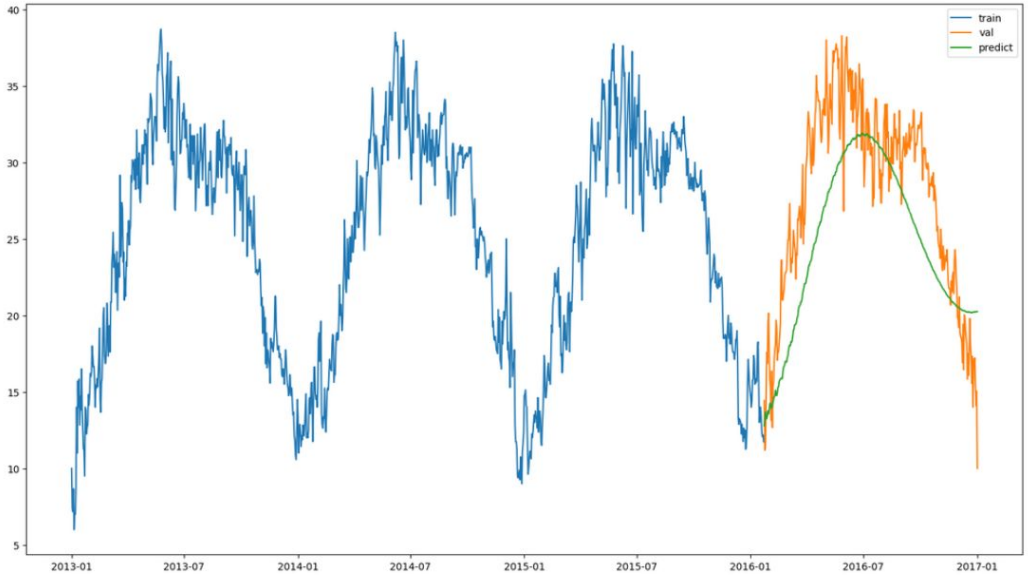


Крутая модель. Но почему она вчера так плохо сработала?

Формула

$$\Delta^q Y_t = c + \sum_{i=1}^{p} \Delta^q Y_{t-i} \alpha_i + \sum_{j=0}^{d} b_j \epsilon_{t-j}$$

График



ARIMA

Крутая модель. Но почему она вчера так плохо сработала?

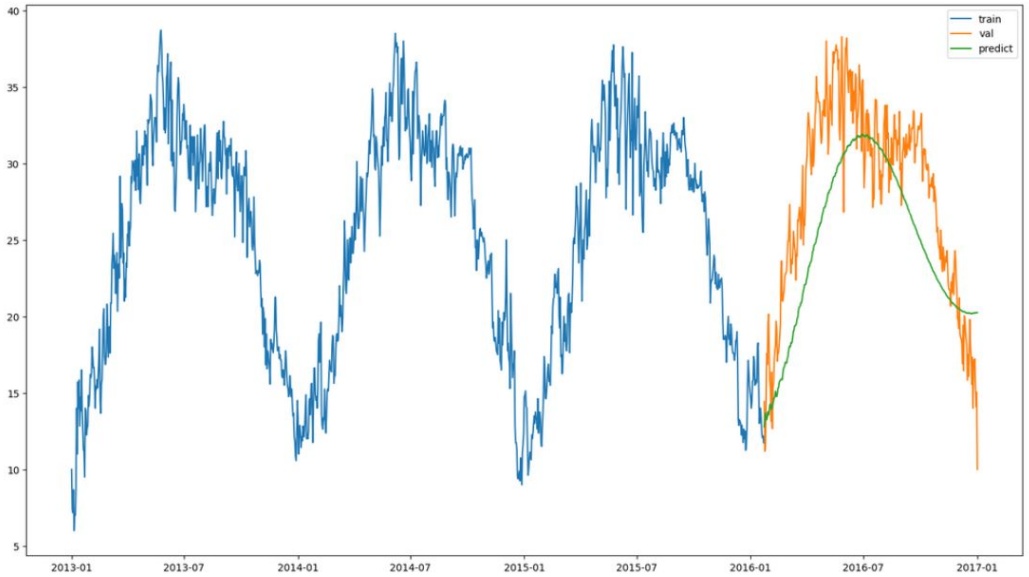
Формула

$$\Delta^q Y_t = c + \sum_{i=1}^{p} \Delta^q Y_{t-i} \alpha_i + \sum_{j=0}^{d} b_j \epsilon_{t-j}$$

Две причины:

- Большой промежуток
- Мало параметров

График



ARIMA

Крутая модель. Но почему она вчера так плохо сработала?

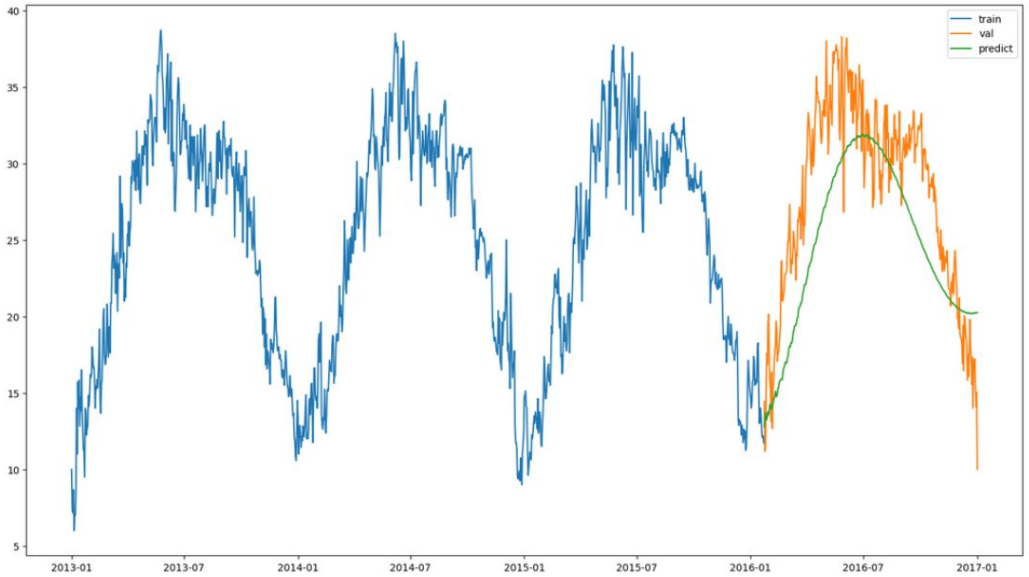
Формула

$$\Delta^q Y_t = c + \sum_{i=1}^{p} \Delta^q Y_{t-i} \alpha_i + \sum_{j=0}^{d} b_j \epsilon_{t-j}$$

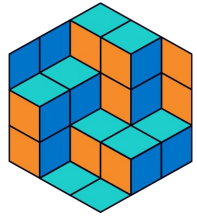
Две причины:

- Большой промежуток
- Мало параметров
- Нет сезонности

График



SARIMA



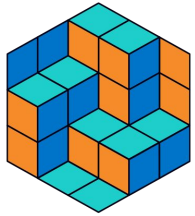
Seasonal ARIMA. Добавим сезонность:

AR(P,S) MA(Q,S)

$$\Delta^d Y_t = c + \sum_{n=1}^{p} \alpha_n \Delta^d Y_{t-n} + \sum_{n=1}^{q} \beta_n \epsilon_{t-n} + \sum_{n=1}^{P} \phi_n \Delta^d Y_{t-sn} + \sum_{n=1}^{Q} \theta_n \epsilon_{t-sn}$$

SARIMA

“Больше параметров - больше шансов сделать лажу”
© Паша Техник

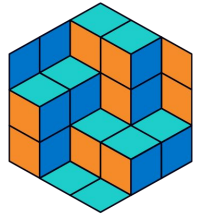


Теперь вместо 3 параметров у нас их 7 (!).

AR(P,S) MA(Q,S)

$$\Delta^d Y_t = c + \sum_{n=1}^{p} \alpha_n \Delta^d Y_{t-n} + \sum_{n=1}^{q} \beta_n \epsilon_{t-n} + \sum_{n=1}^{P} \phi_n \Delta^d Y_{t-sn} + \sum_{n=1}^{Q} \theta_n \epsilon_{t-sn}$$

SARIMA

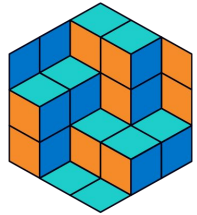


Обучение: Все тот же **MSE**.

AR(P,S) MA(Q,S)

$$\Delta^d Y_t = c + \sum_{n=1}^{p} \alpha_n \Delta^d Y_{t-n} + \sum_{n=1}^{q} \beta_n \epsilon_{t-n} + \sum_{n=1}^{P} \phi_n \Delta^d Y_{t-sn} + \sum_{n=1}^{Q} \theta_n \epsilon_{t-sn}$$

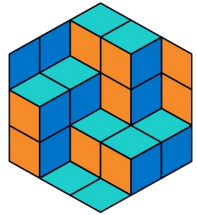
SARIMA Pro MAX Ultra-Wide



Но можно ведь сделать еще больше параметров?

Из 7 сделать, например… сколько захотим!

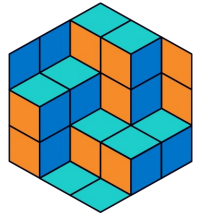
SARIMAX



Такая модель называется **SARIMAX**.

Ее отличие (да, вы угадали) – еще параметры!

SARIMAX

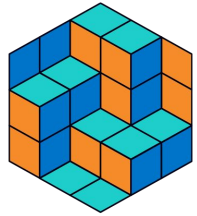


Такая модель называется **SARIMAX**.

Ее отличие (да, вы угадали) – еще параметры!

Внешние параметры

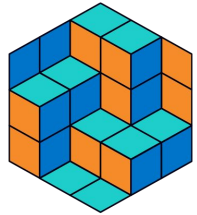
$$\Delta^d Y_t = c + \sum_{n=1}^{p} \alpha_n \Delta^d Y_{t-n} + \sum_{n=1}^{q} \beta_n \epsilon_{t-n} + \sum_{n=1}^{P} \phi_n \Delta^d Y_{t-sn} + \sum_{n=1}^{Q} \theta_n \epsilon_{t-sn} + \sum_{n=1}^{R} \psi_n X_{n_t}$$



Глава 2

Градиентный Бустинг

Хехе

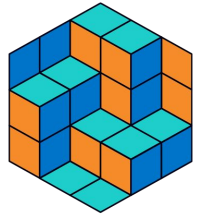


Помните регрессии? Нет, они не закончились :)

Более того, во временных рядах они встречаются даже в очень крутых моделях!

На (еще) одну из таких сейчас посмотрим.

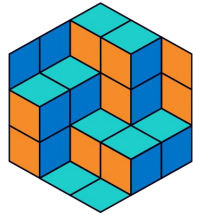
Градиентный Бустинг. Интро



Рассмотрим задачу линейной регрессии с минимизацией MSE.

$$\mathcal{L}(y, x) = \frac{1}{2} \sum_{i=1}^{N} (y_i - a(x_i))^2 \to \min$$

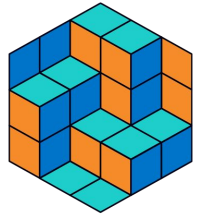
Градиентный Бустинг. Интро



Представим, что мы построили одну такую модель (плохую, конечно)

Она на каком-то значении дает ошибку ***10***

Градиентный Бустинг. Интро

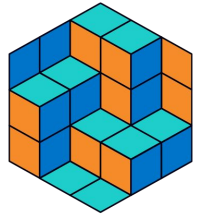


Представим, что мы построили одну такую модель (плохую, конечно)

Она на каком-то значении дает ошибку ***10***

Ну давайте построим еще одну, которая будет нивелировать эту ошибку - то есть на этом значении давать ***-10***

Градиентный Бустинг. Интро



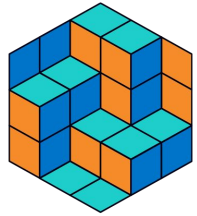
Представим, что мы построили одну такую модель (плохую, конечно)

Она на каком-то значении дает ошибку ***10***

Ну давайте построим еще одну, которая будет нивелировать эту ошибку - то есть на этом значении давать ***-10***

Очевидно, идеально не получится - где-то вылезет еще одна ошибка

Градиентный Бустинг. Интро



Представим, что мы построили одну такую модель (плохую, конечно)

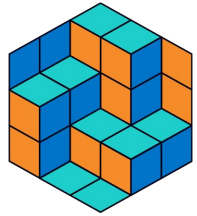
Она на каком-то значении дает ошибку ***10***

Ну давайте построим еще одну, которая будет нивелировать эту ошибку - то есть на этом значении давать ***-10***

Очевидно, идеально не получится - где-то вылезет еще одна ошибка

Не беда - построим еще одну регрессию, чтобы она нивелировала ошибки двух предыдущих регрессий.

Градиентный Бустинг. Интро



Представим, что мы построили одну такую модель (плохую, конечно)

Она на каком-то значении дает ошибку ***10***

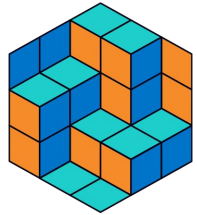
Ну давайте построим еще одну, которая будет нивелировать эту ошибку - то есть на этом значении давать ***-10***

Очевидно, идеально не получится - где-то вылезет еще одна ошибка

Не беда - построим еще одну регрессию, чтобы она нивелировала ошибки двух предыдущих регрессий.

И так далее

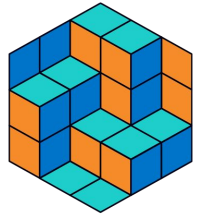
Градиентный Бустинг. Что это?



Такой метод называется **Градиентный бустинг**.

Кстати говоря, он вообще широко используется - в регрессиях, классификации.

Градиентный Бустинг. Что это?



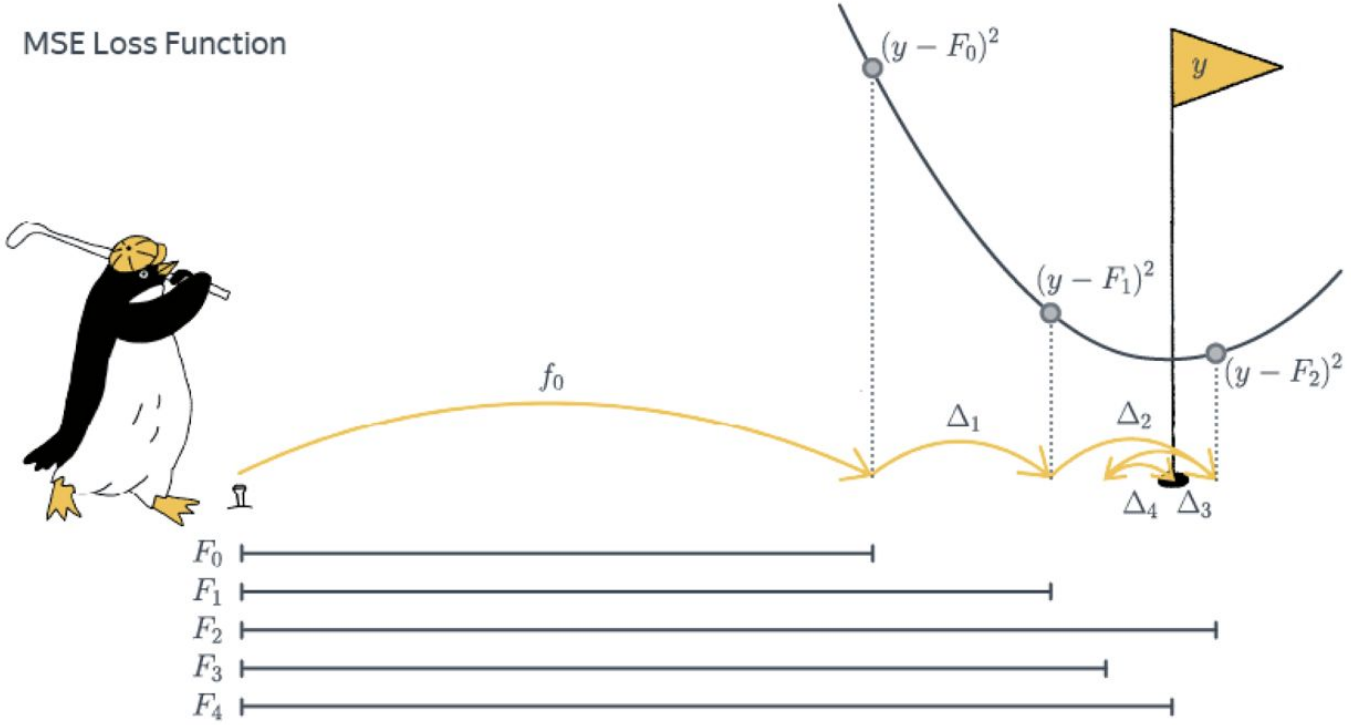
Такой метод называется **Градиентный бустинг**.

Кстати говоря, он вообще широко используется - в регрессиях, классификации.

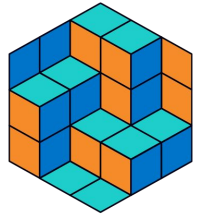
Формально, мы строим такую модель:

$$a(x) = a_K(x) = b_1(x) + b_2(x) + \cdots + b_K(x)$$

Градиентный Бустинг. Картинка из интернета



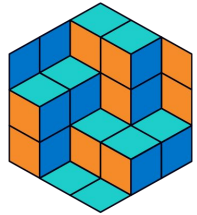
Градиентный Бустинг. Что это?



Отличие от обычной регрессии – мы подбираем не все регрессии разом, а по очереди.

$$a(x) = a_K(x) = b_1(x) + b_2(x) + \cdots + b_K(x)$$

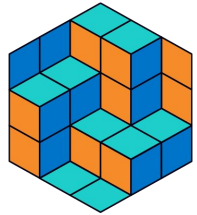
Градиентный Бустинг. Как работает



Сначала мы обучаем первый алгоритм:

$$b_1(x) = \underset{b \in \mathcal{B}}{\operatorname{argmin}}, \mathcal{L}(y, b(x))$$

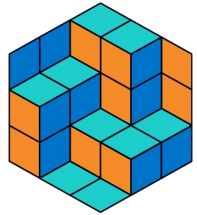
Градиентный Бустинг. Как работает



Потом мы находим разницу истинных значений и предсказанных первым алгоритмом:

$$s_i^1 = y_i - b_1(x_i)$$

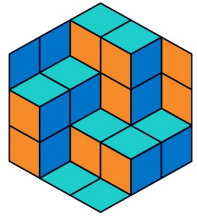
Градиентный Бустинг. Как работает



Теперь мы хотим обучить второй классификатор, но уже чтобы он предсказывал разницу:

$$b_2(x) = \underset{b \in \mathcal{B}}{\operatorname{argmin}}, \mathcal{L}(s^1, b(x))$$

Градиентный Бустинг. Как работает



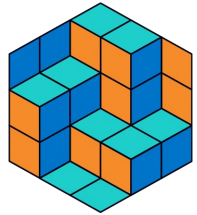
И так далее.

Будет круто, если второй уже предскажет все правильно, ведь тогда:

$$a_2(x_i) = b_1(x_i) + b_2(x_i) =$$

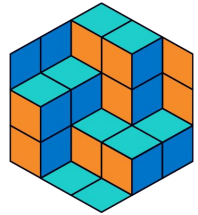
$$= b_1(x_i) + s_i^1 = b_1(x_i) + (y_i - b_1(x_i)) = y_i$$

Градиентный Бустинг. Аутро



А только ли регрессии там могут использоваться?

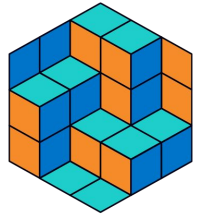
Градиентный Бустинг. Аутро



А только ли регрессии там могут использоваться?

Нет, вообще любые не тяжелые модельки - например, ***Решающие деревья***.

Замечания

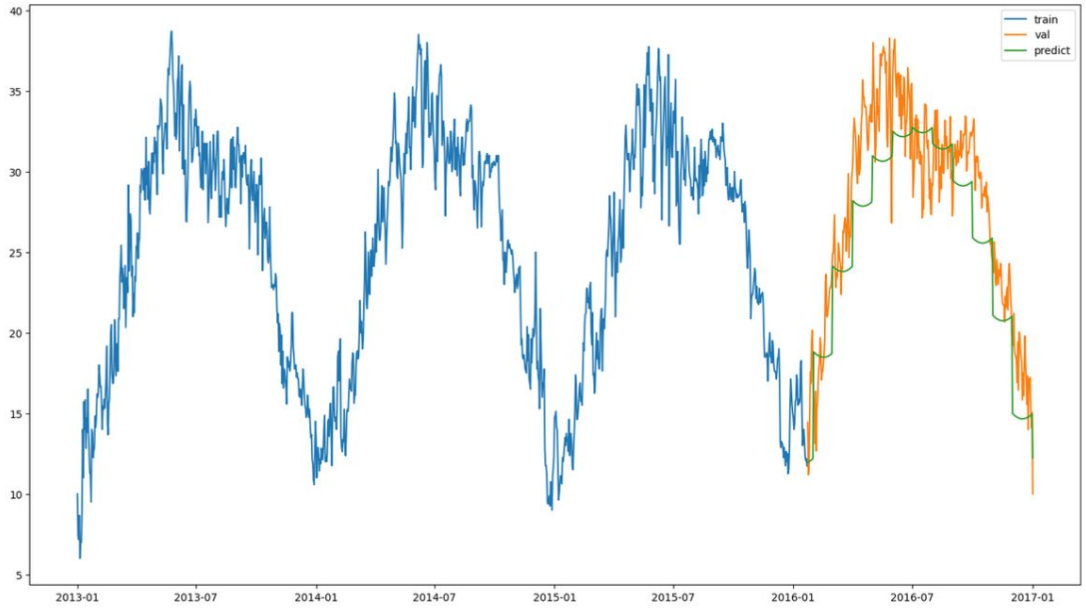


Может получиться так, что первый алгоритм уже хорош - тогда мы можем (совершенно случайно!) переобучиться. За этим надо следить!

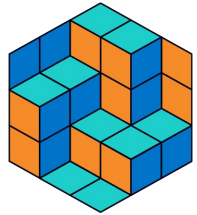
Замечание

А помните линейную регрессию на стероидах с первой пары?

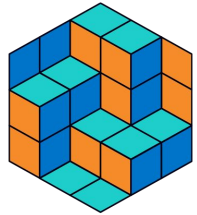
$$\sum_{n=1}^{R} \psi_n X_{n_t}$$



Замечание



Получается, что помимо крутой модели можно сделать еще и очень крутые данные?



Глава 3

Feature Engineering

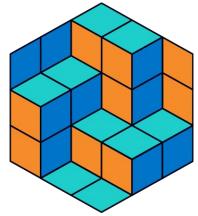
Feature Engineering

Feature Engineering - одна из задач во всем известном Data Science.

Главная задача - сделать данные лучше.

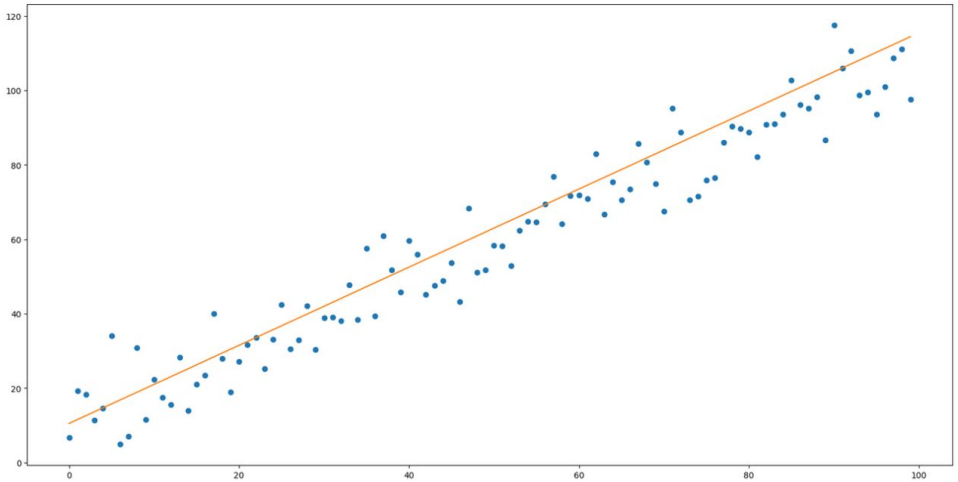
Методов - море. Сейчас посмотрим на некоторые из них.

Пример

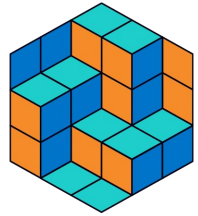


Есть данные ***(x,y)***, хотим их предсказать.

Метод: строим регрессию:

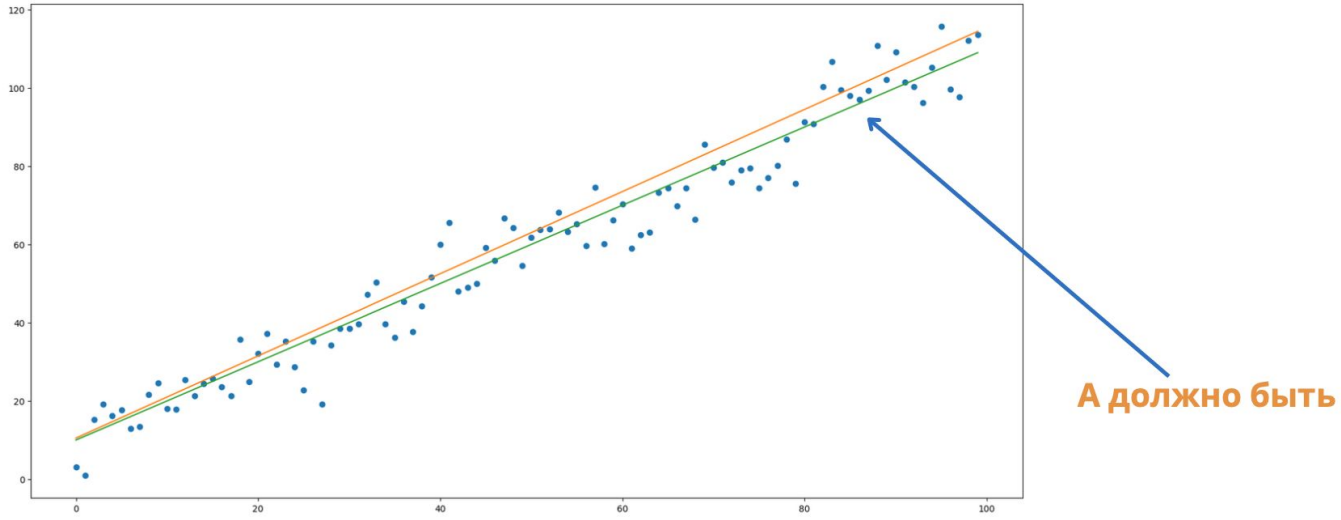


Пример



Есть данные ***(x,y)***, хотим их предсказать.

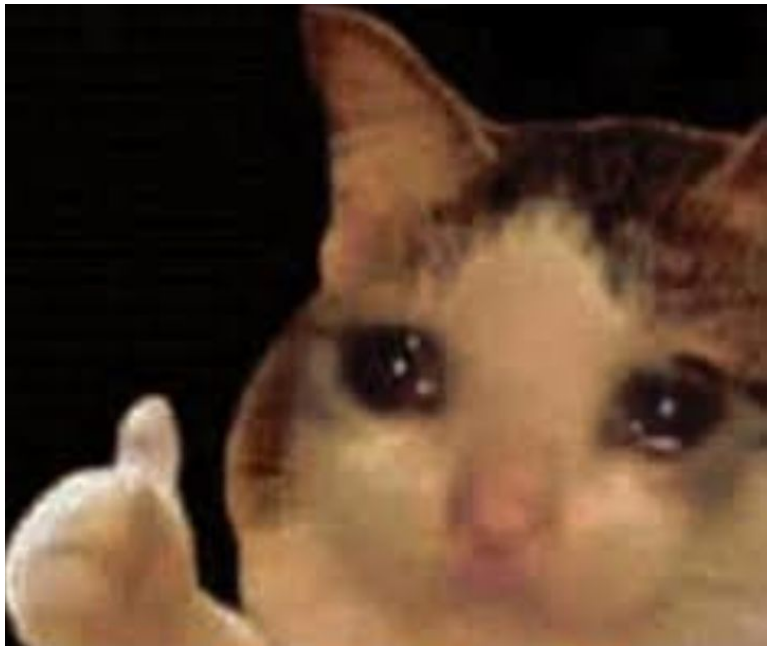
Метод: строим регрессию:



Пример

Дальше веса не идут, MSE не меняется.

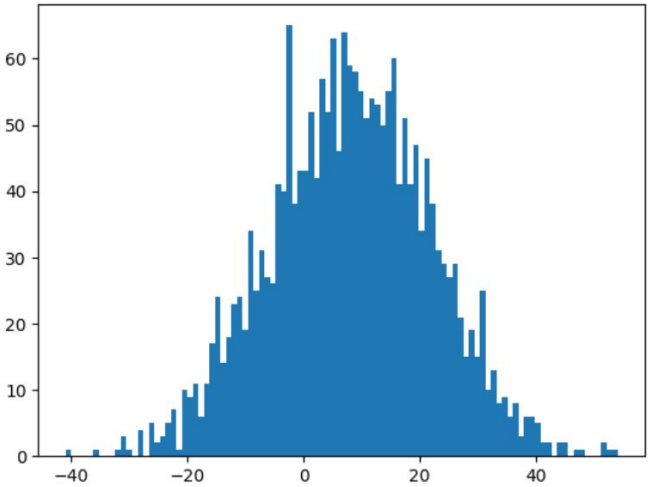
Что делать?



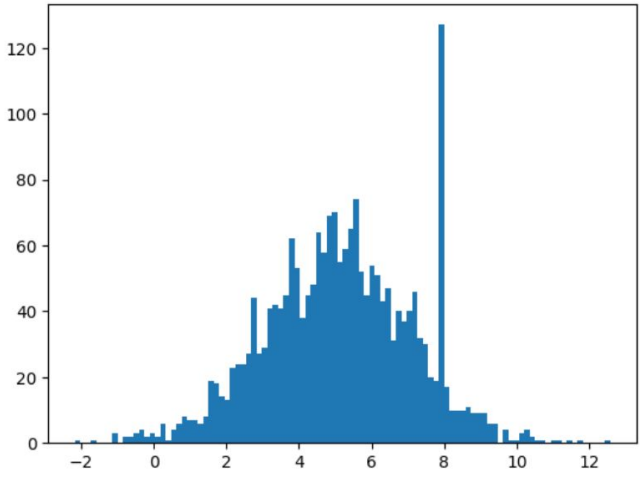
Гистограммы

Давайте посмотрим на распределения наших параметров:

a

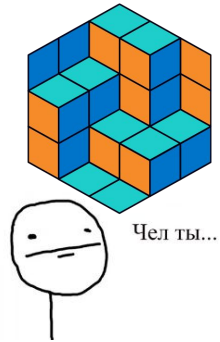


b

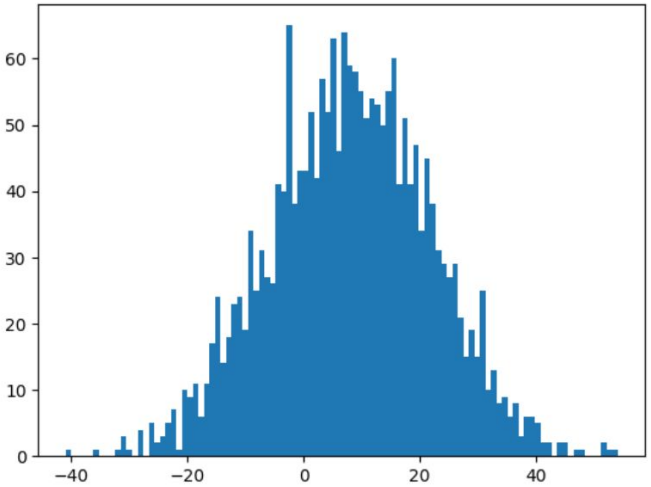


Гистограммы

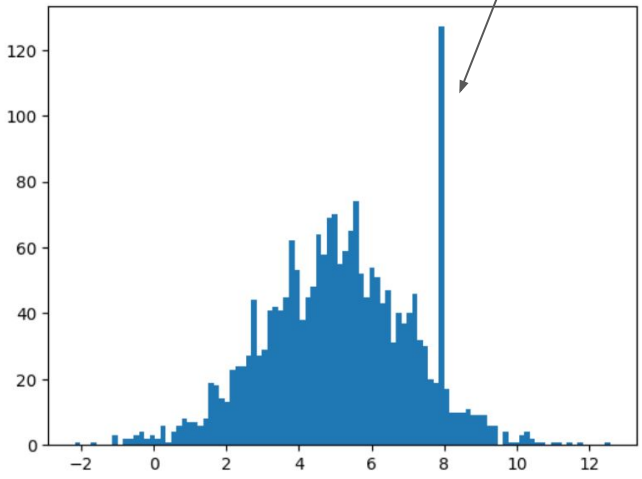
Проблема очевидна: в b есть тупые значения, которые все портят!



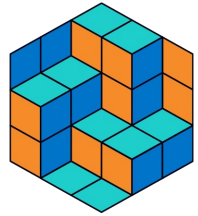
a



b

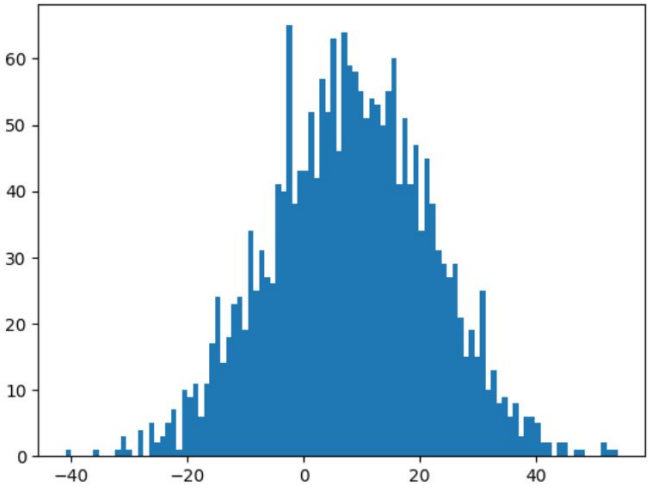


Гистограммы

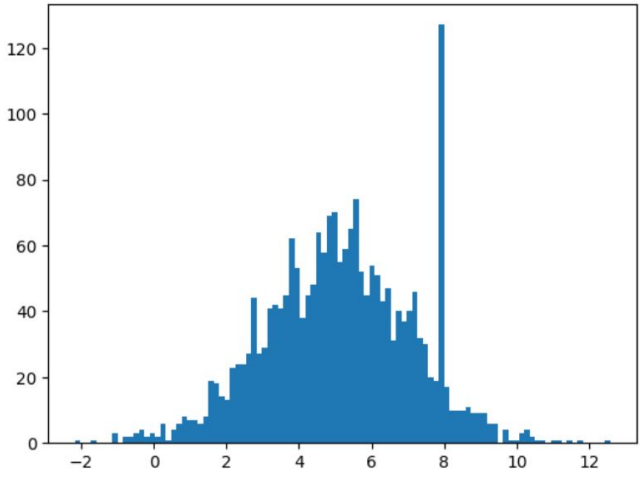


Проблема очевидна: в b есть тупые значения, которые все портят! *Уберем их*

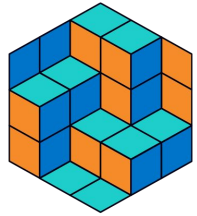
a



b

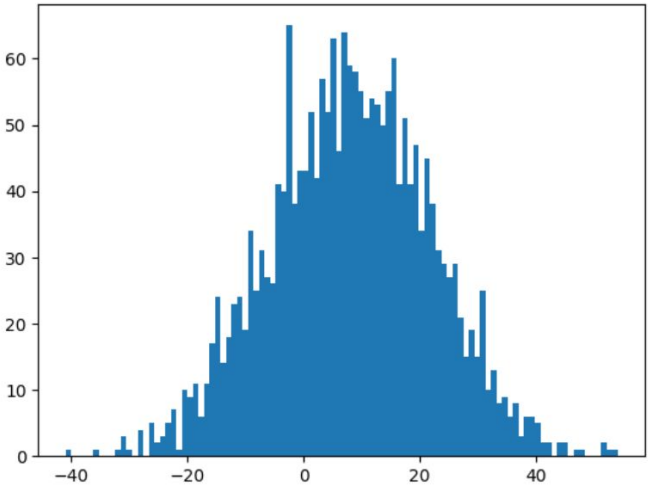


Выбросы

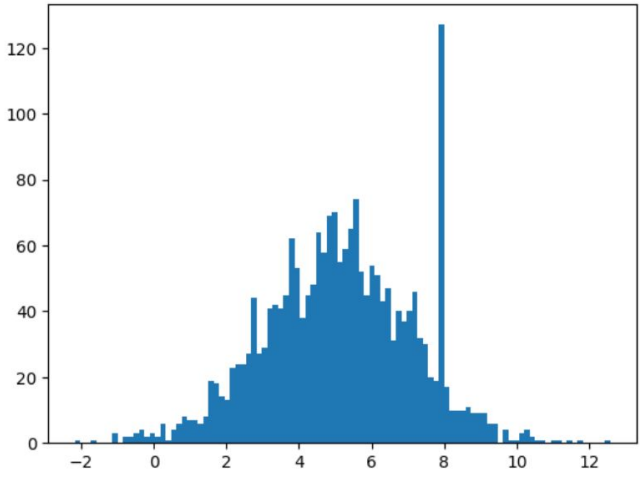


Такие штуки называются ***выбросами*** - они могут быть как в значениях, так и в параметрах.

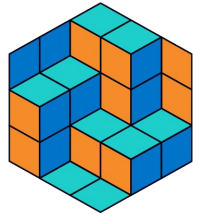
a



b

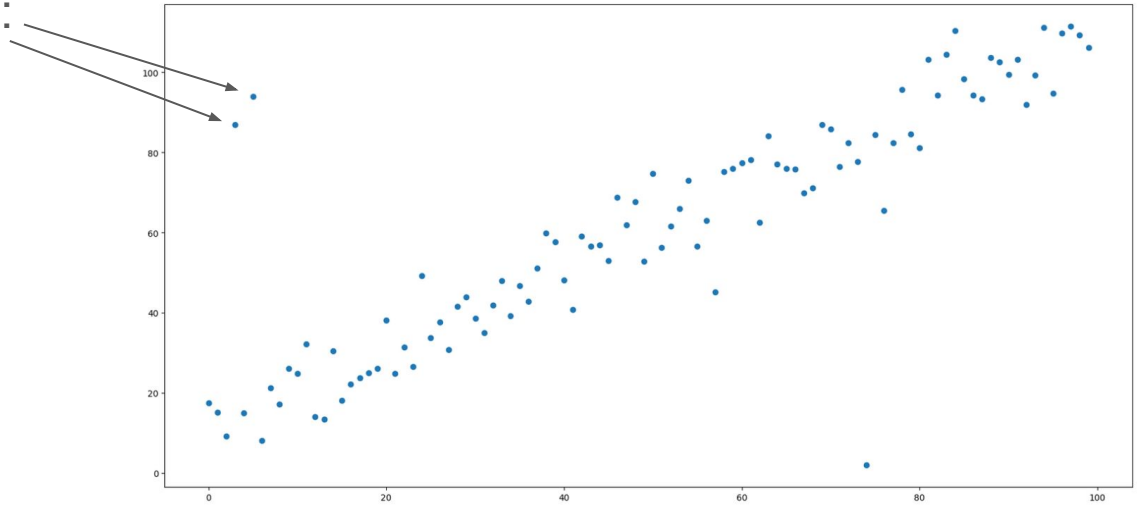


Выбросы

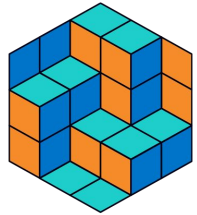


Такие штуки называются ***выбросами*** - они могут быть как в значениях, так и в параметрах.

В значениях - например вот:



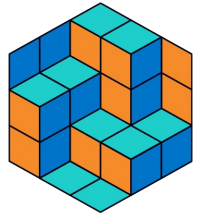
Выбросы



Методы поиска выбросов:

- Квантили
- DBScan/K-Means, или другие ***кластеризации***
- BoxPlot

Выбросы



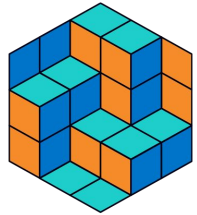
Методы поиска выбросов:

- Квантили
- DBScan/K-Means, или другие ***кластеризации***
- BoxPlot

Что с ними делать?

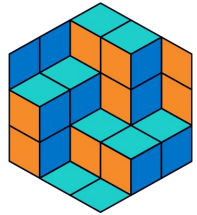
- Удалить (почему нет)
- Заменить средним по выборке (не всегда хорошо)
- Заменить средним по окну (чуть получше, особенно в рядах)

Метод 2. Генерация параметров



Помните как мы делали параметры в линейной регрессии? Как они назывались?

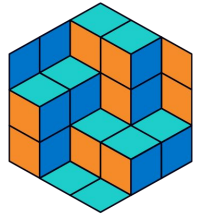
Метод 2. Генерация параметров



Помните как мы делали параметры в линейной регрессии? Как они назывались? Верно, ***экзогенные переменные***

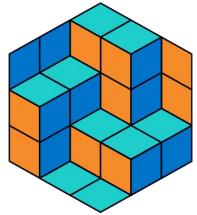
Можно просто добавлять их. А есть какие-то методы?

Метод 2. Генерация параметров



Есть. Пусть у нас есть линейная регрессия **y = pa + qb**. Имеет ли смысл добавлять параметр **(a+b)**?

Метод 2. Генерация параметров

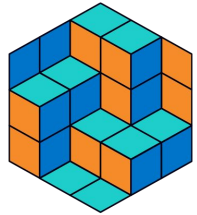


Есть. Пусть у нас есть линейная регрессия **y = pa + qb**. Имеет ли смысл добавлять параметр **(a+b)**?

Нет конечно, ведь **y= pa+qb = p'a + q'b + u(a+b)**

А все потому что ***линейные комбинации*** и так строятся в линейной регрессии.

Метод 2. Генерация параметров



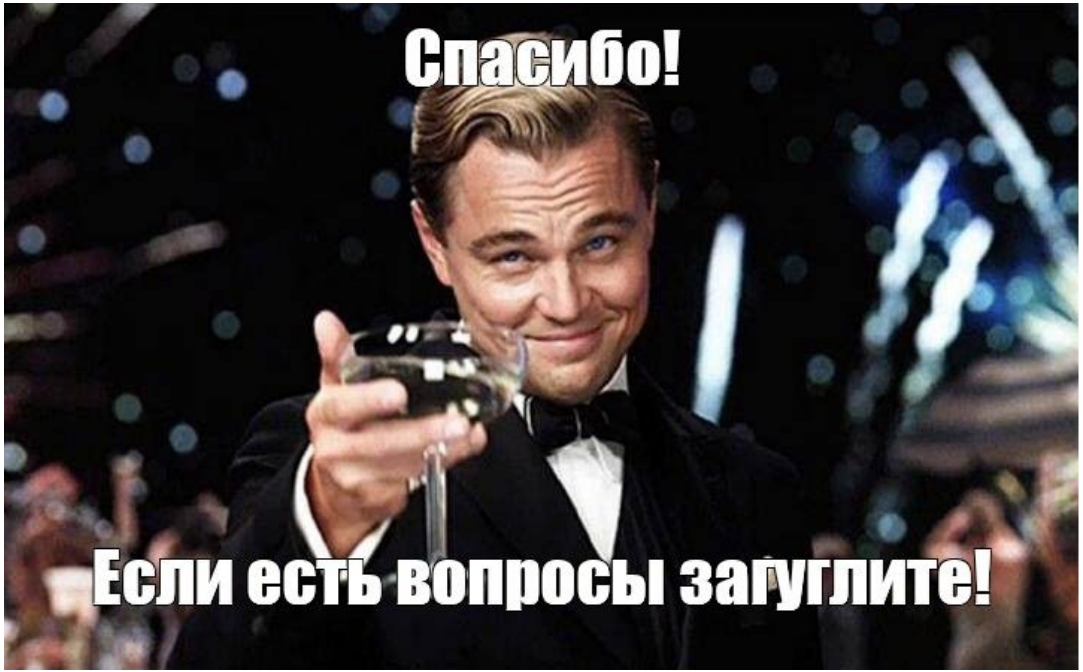
Есть. Пусть у нас есть линейная регрессия **y = pa + qb**. Имеет ли смысл добавлять параметр **(a+b)**?

Нет конечно, ведь **y= pa+qb = p'a + q'b + u(a+b)**

А все потому что ***линейные комбинации*** и так строятся в линейной регрессии.

Поэтому топ-стратегия - делать нелинейные комбинации – **a*b**, **a*a**, и т.д.

Вопросы?



Соревнование!!!

А теперь устроим соревнование:

- Нужна будет Практика 3
- Нужен бот в телеге - @sirius_ts_bot – через него сдаем ваши предсказания, там же смотрим топ!