# Знакомство с классификацией на примере логистической регрессии

Никита

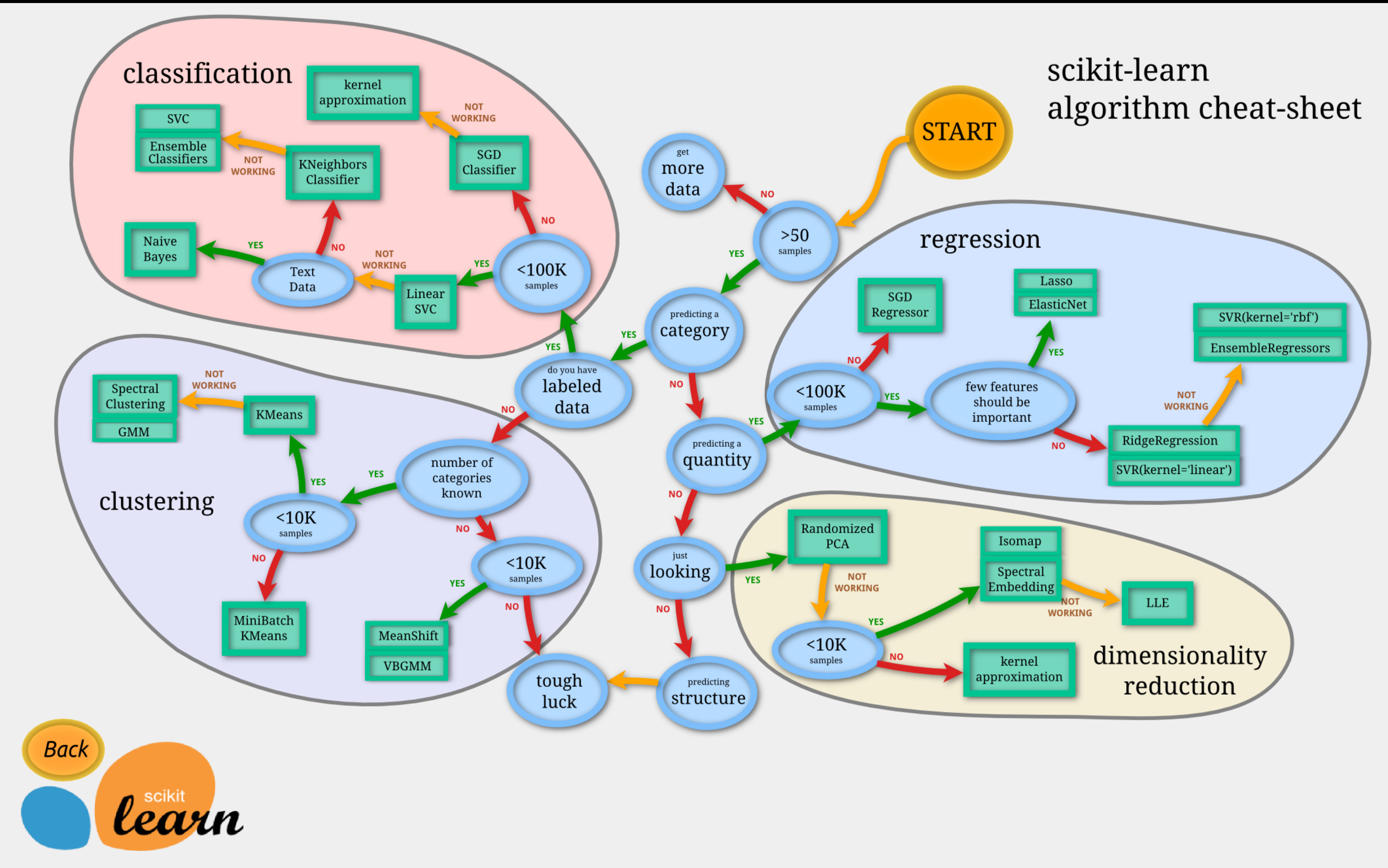
# Сегодня в программе

- Задачи машинного обучения
- Регрессия
- Классификация

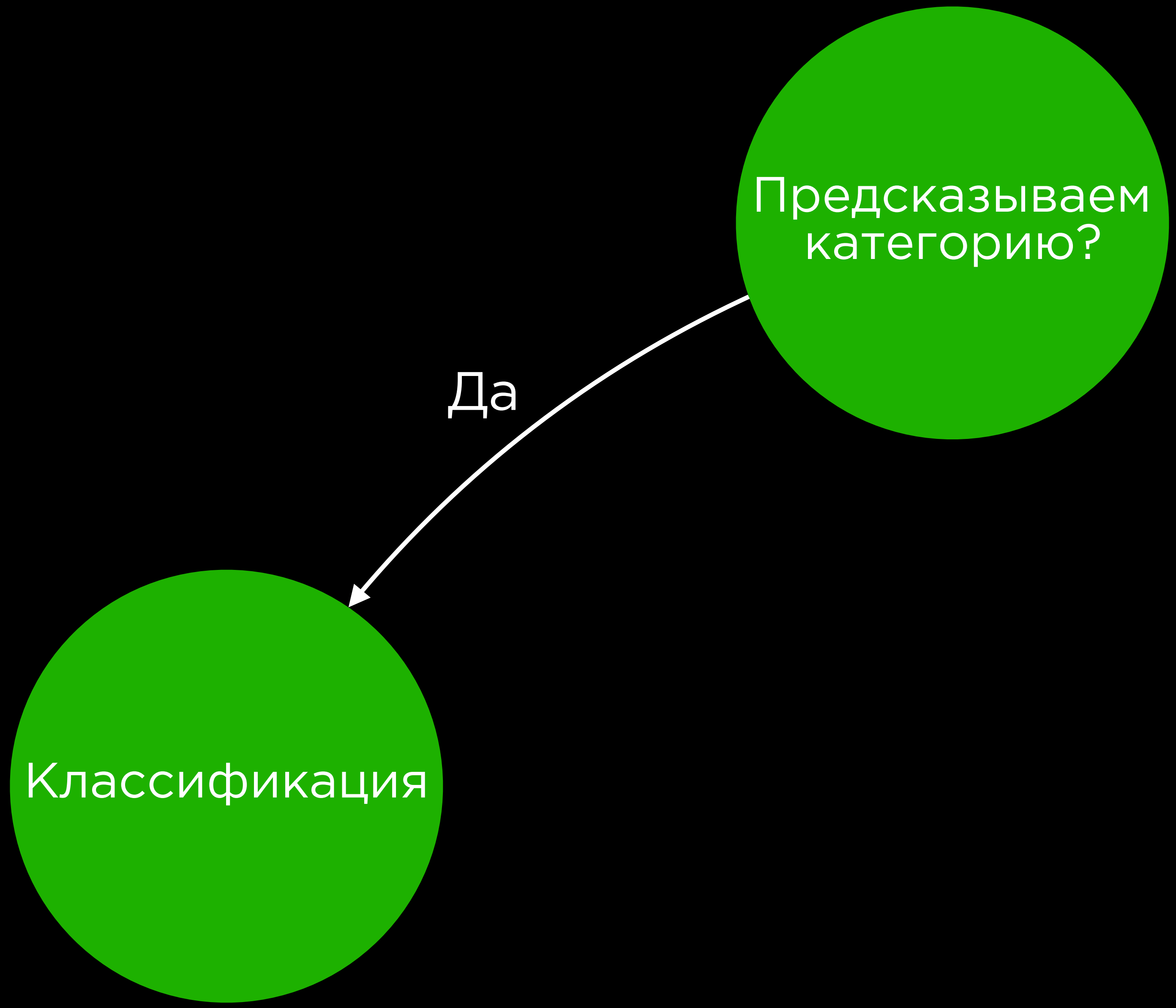
# Сегодня в программе

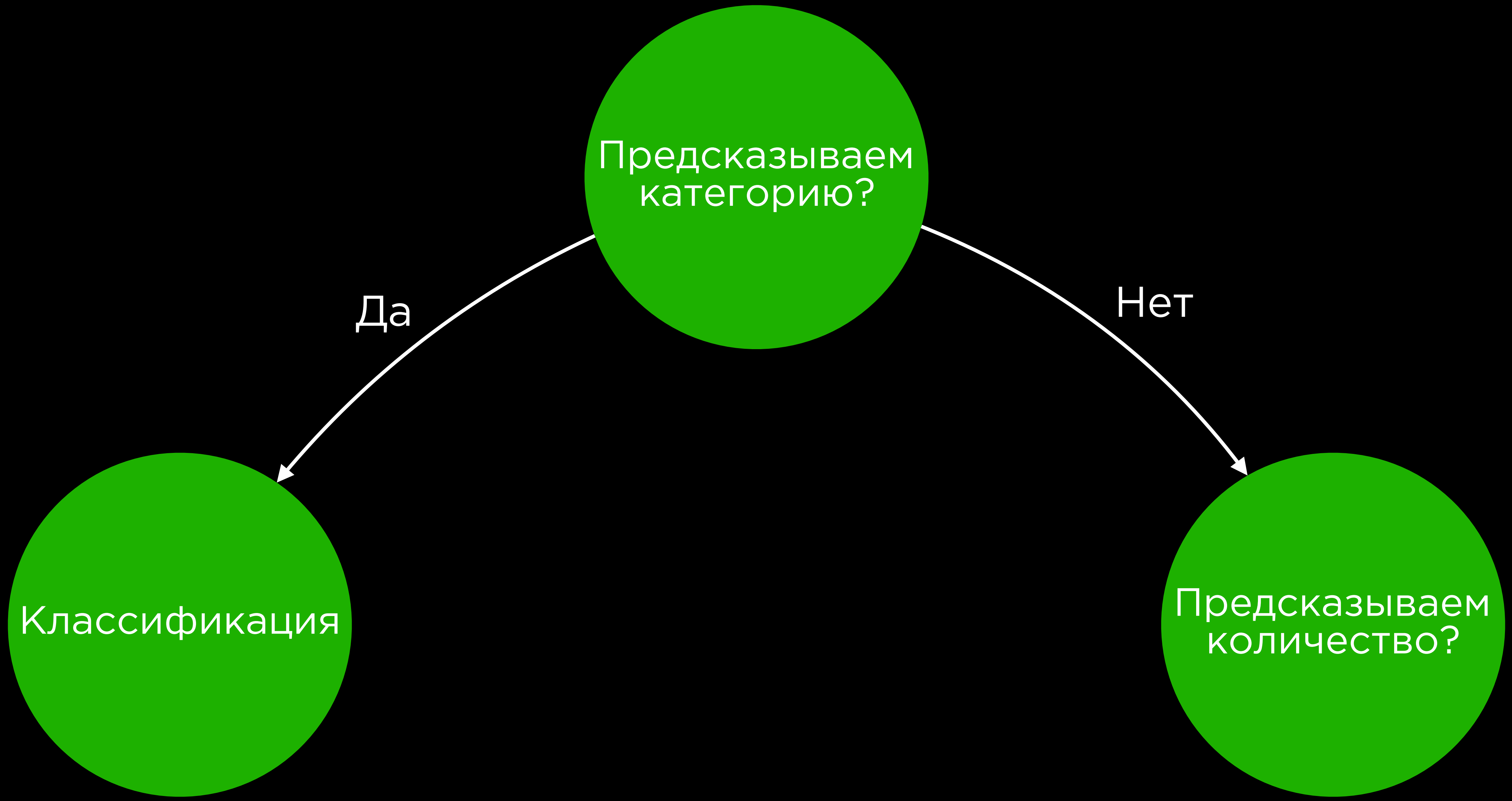
- Задачи машинного обучения
- Регрессия
    - Линейная регрессия
        - МНК оценки
        - Псевдообратная матрица
        - Градиентный спуск
        - Стохастический градиентный спуск
    - Как оценить качество регрессии
- Классификация
    - От непрерывного к дискретному
    - Все дело в функции потерь
    - Как оценить качество классификации?

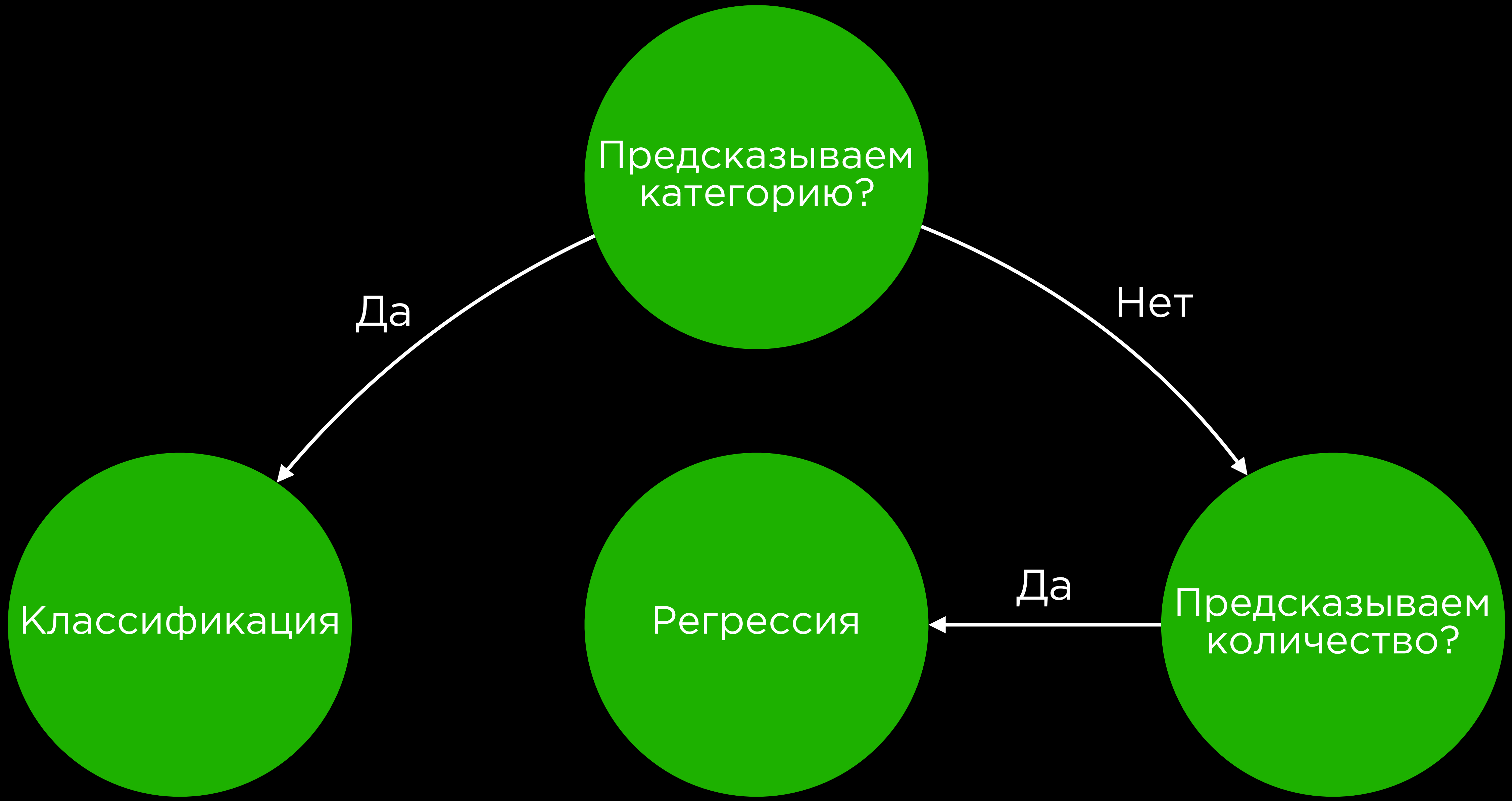
# Задачи машинного обучения

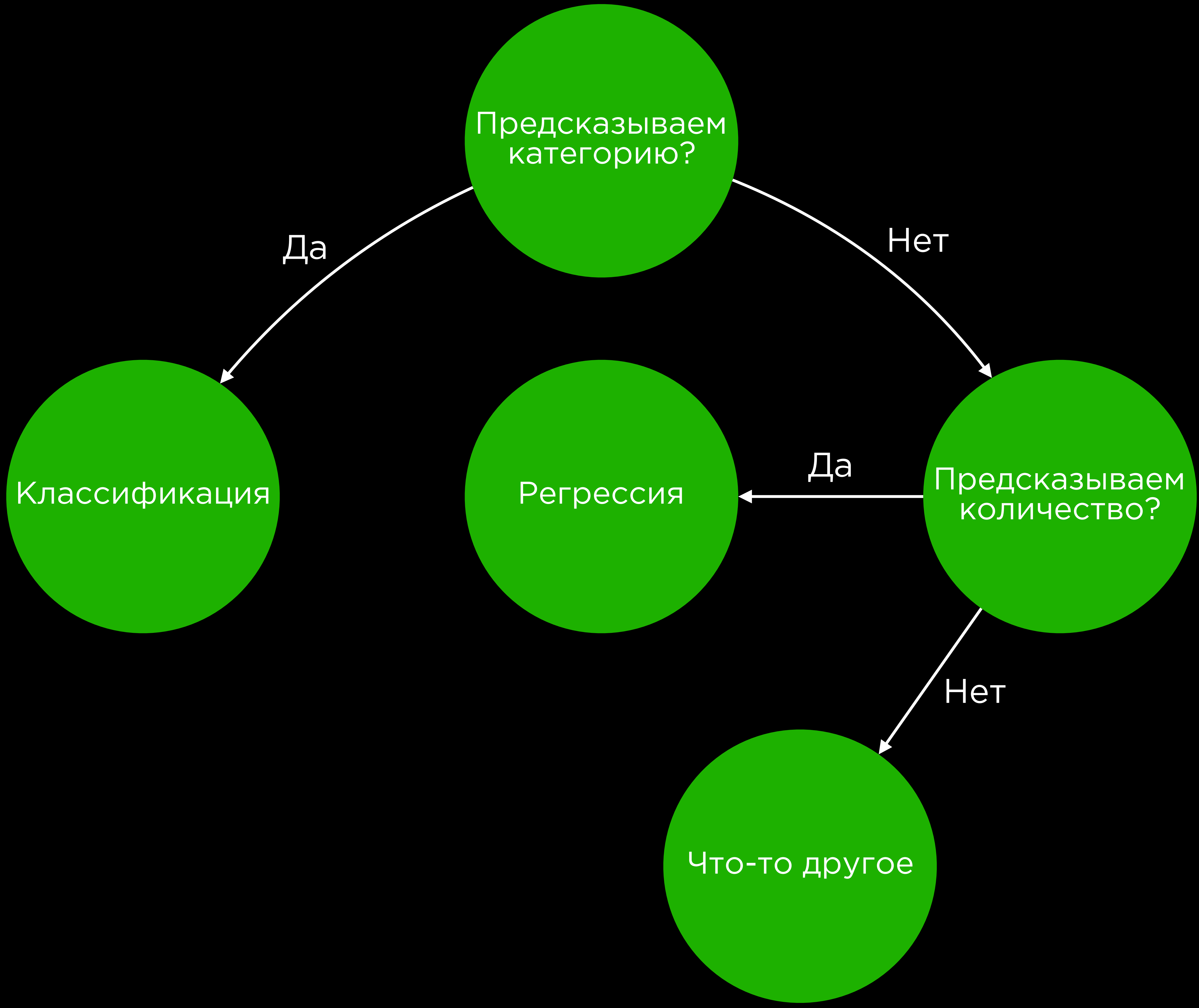

scikit-learn
algorithm cheat-sheet
START
>50 samples
get more data
NO
YES
predicting a category
do you have labeled data
predicting a quantity
just looking
predicting structure
tough luck
classification
kernel approximation
SVC
Ensemble Classifiers
KNeighbors Classifier
SGD Classifier
Naive Bayes
Text Data
Linear SVC
<100K samples
NOT WORKING
clustering
Spectral Clustering
GMM
KMeans
<10K samples
MiniBatch KMeans
MeanShift
VBGMM
number of categories known
regression
SGD Regressor
Lasso
ElasticNet
SVR(kernel='rbf')
EnsembleRegressors
few features should be important
RidgeRegression
SVR(kernel='linear')
dimensionality reduction
Randomized PCA
Isomap
Spectral Embedding
LLE
Back
scikit learn

Предсказываем категорию?


Предсказываем категорию?
Да
Классификация


Предсказываем категорию?
Да
Нет
Классификация
Предсказываем количество?


Предсказываем категорию?
Да
Нет
Классификация
Регрессия
Да
Предсказываем количество?


Предсказываем категорию?
Да
Нет
Классификация
Регрессия
Да
Предсказываем количество?
Нет
Что-то другое


Предсказываем категорию?
Да
Нет
Классификация
Регрессия
Да
Предсказываем количество?
Нет
Что-то другое

# Регрессия

# Постановка задачи

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

# Постановка задачи

$$Y = f(\underline{X_1, \ldots, X_l}) + \epsilon$$

# Постановка задачи

$$Y = f(\underline{X_1, \ldots, X_l}) + \epsilon$$

Независимые
переменные

# Постановка задачи

$$Y = f(\underbrace{X_1, \ldots, X_l}_{\text{Независимые переменные}}) + \epsilon$$

Целевое значение

Независимые переменные

# Постановка задачи

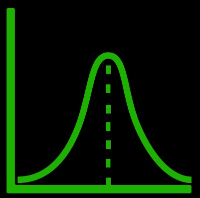
Функция

$$Y = f(\underline{X_1, \ldots, X_l}) + \epsilon$$

Целевое значение

Независимые переменные

# Постановка задачи



Функция

Нормальный шум

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

Целевое значение

Независимые переменные

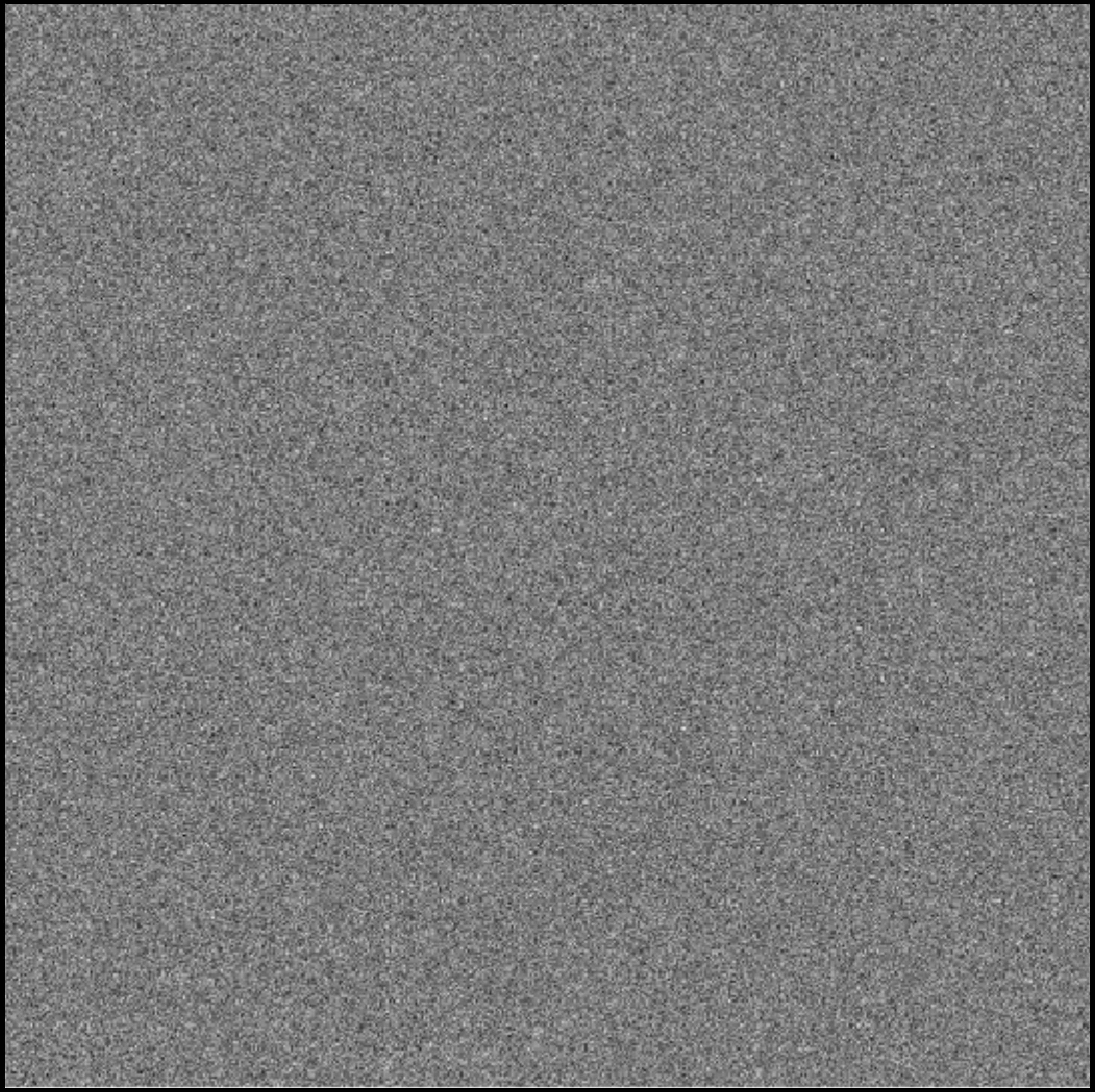
# Самое простое F?

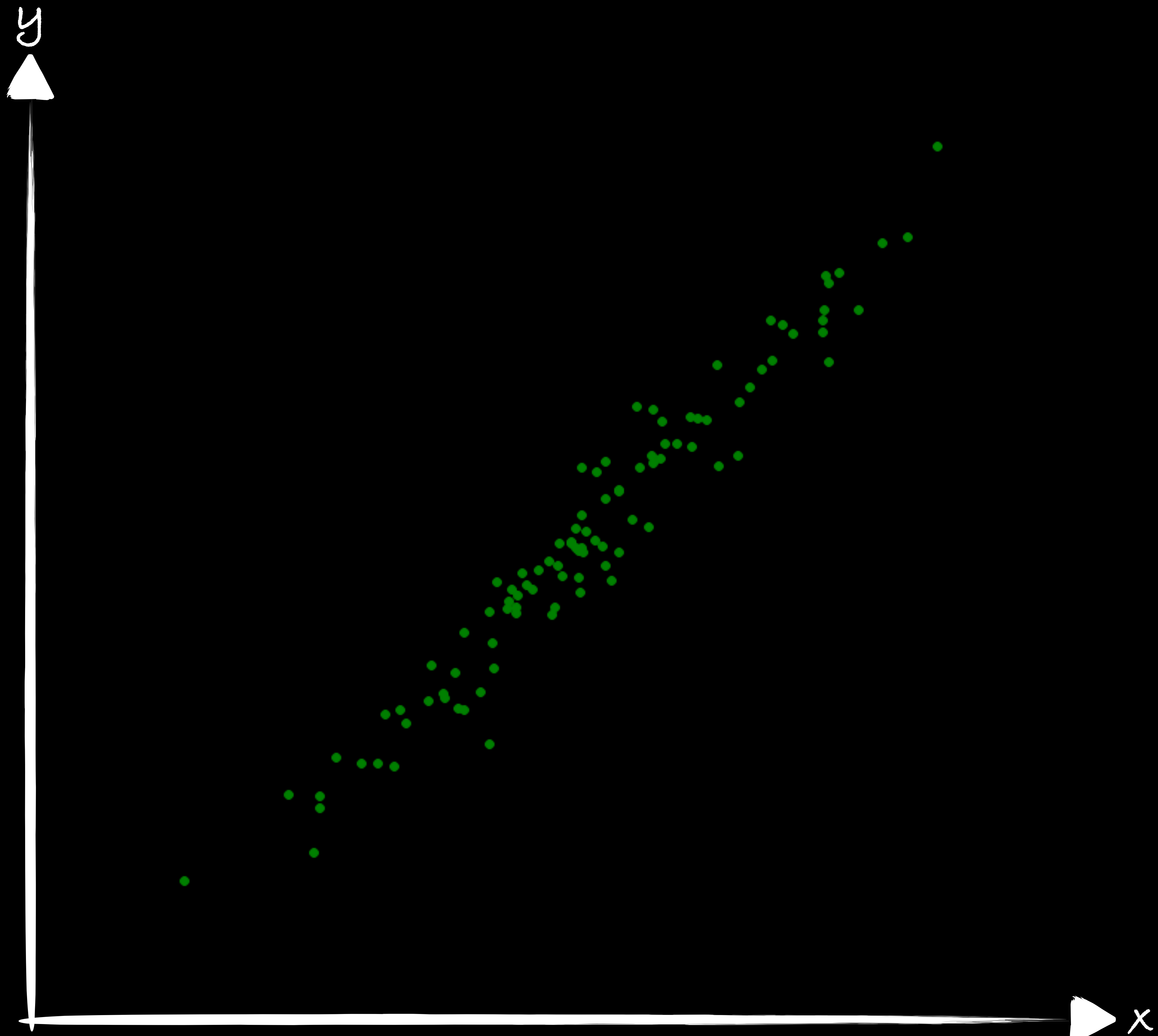
# Константа!

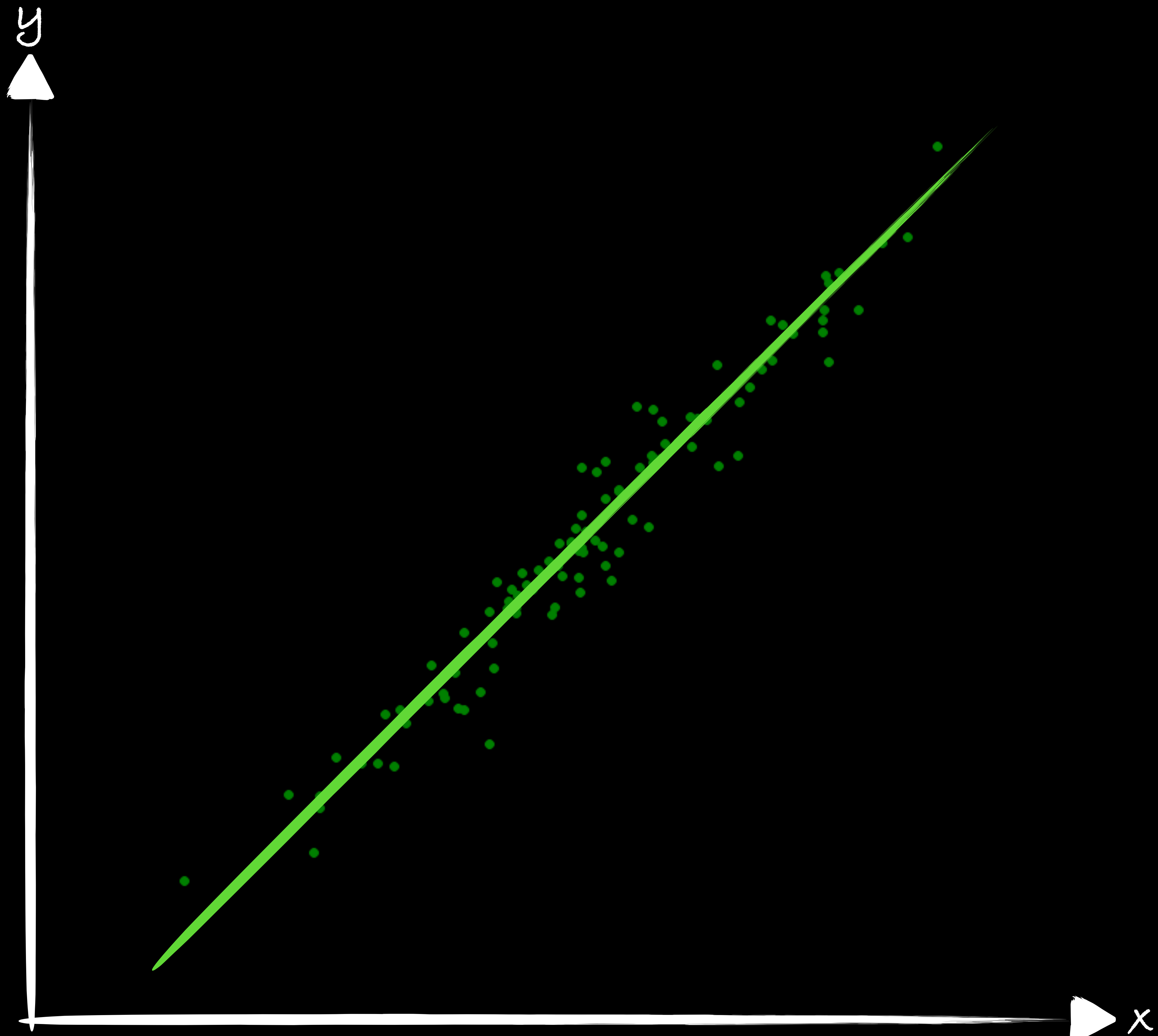
$$f(X) = 5$$

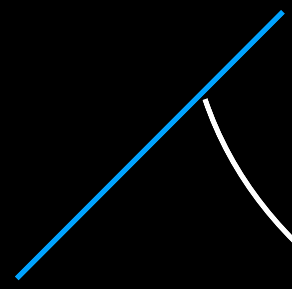
Сложнее, но также бесполезно?

$$f(X) =$$

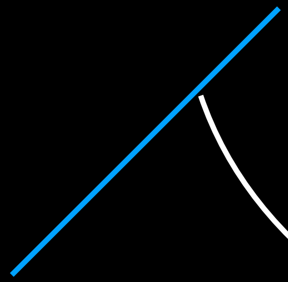



y
x

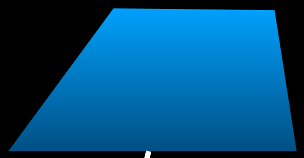

y
x



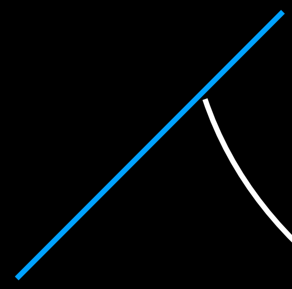
$$y = ax + b$$



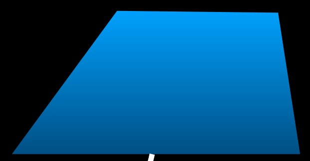
$$y = ax + b$$



$$y = a_1x_1 + a_2x_2 + b$$

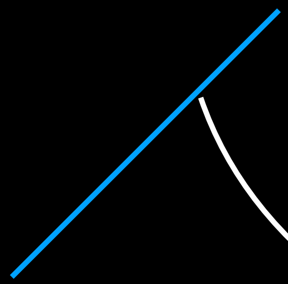


$$y = ax + b$$

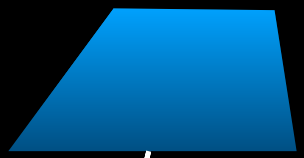


$$y = a_1 x_1 + a_2 x_2 + b$$

$$\cdots$$



$$y = ax + b$$



$$y = a_1x_1 + a_2x_2 + b$$

$$\cdots$$

$$Y = Xa + b$$

Плоскость в N-мерном пространстве

# Магистратура

Магистратура

Магистратура

Ожидаемые баллы

Магистратура

Ожидаемые баллы

Результаты прошлых лет

Магистратура

Ожидаемые баллы

Результаты прошлых лет

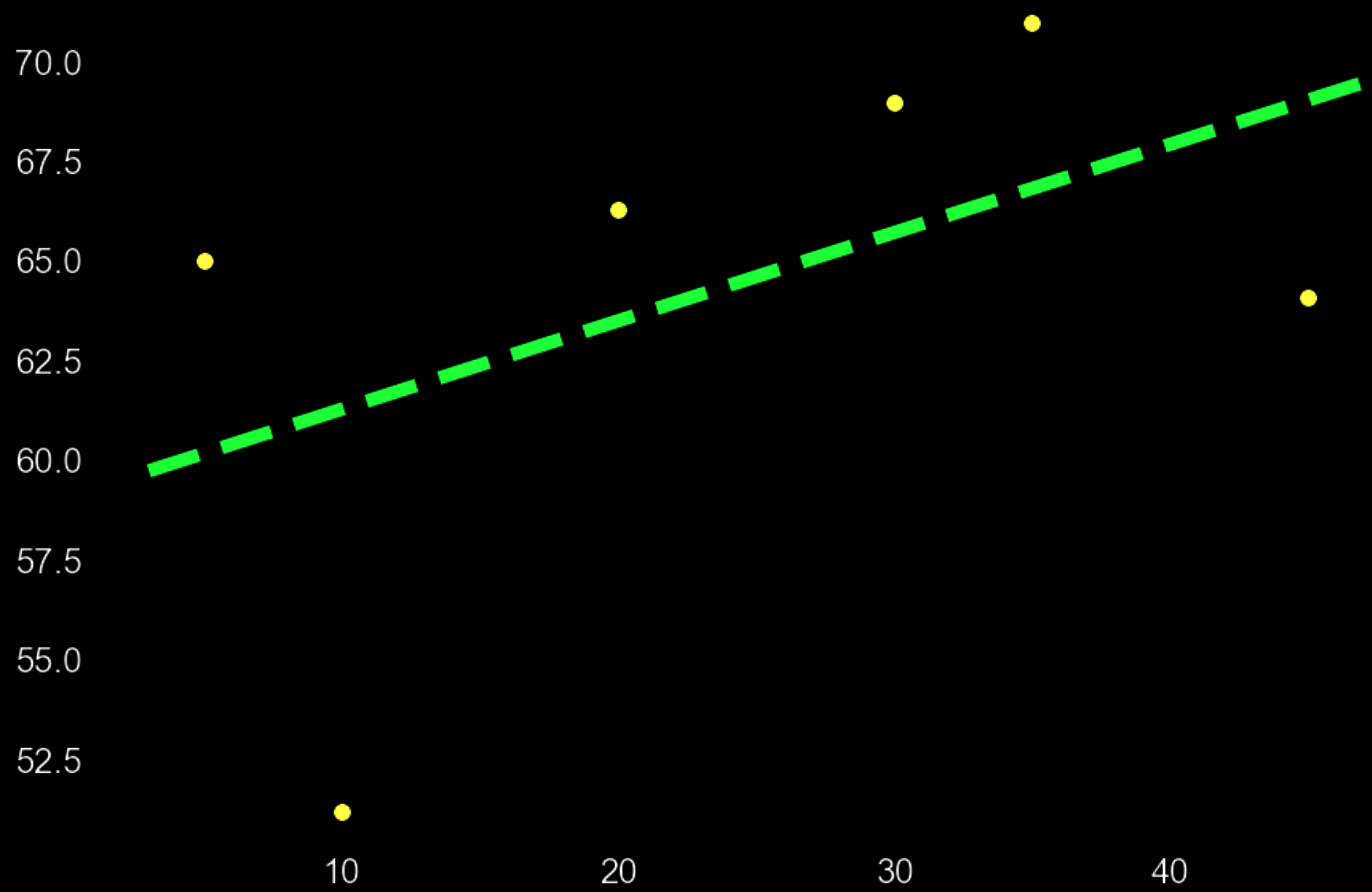
Количество мест

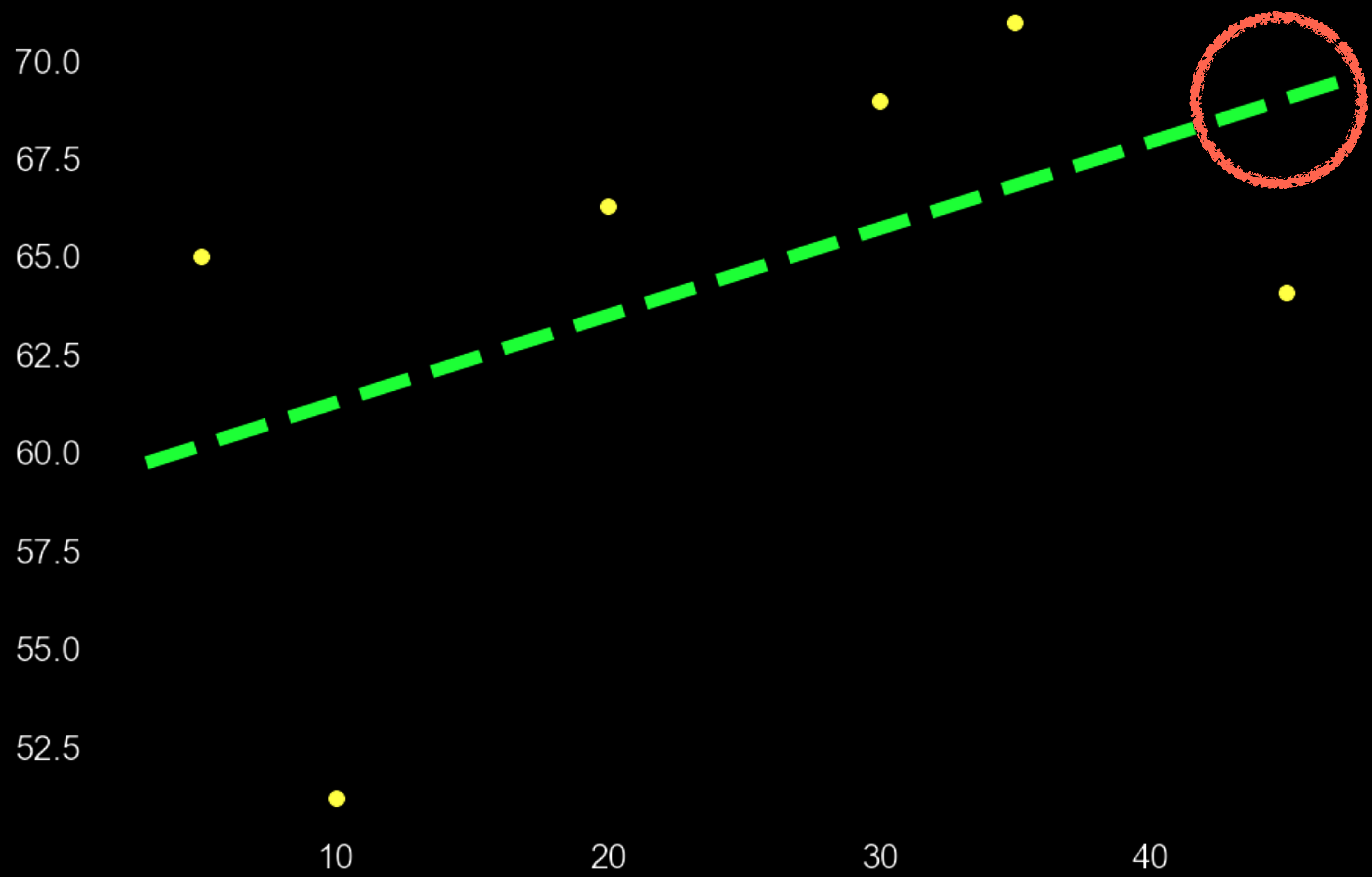
Магистратура

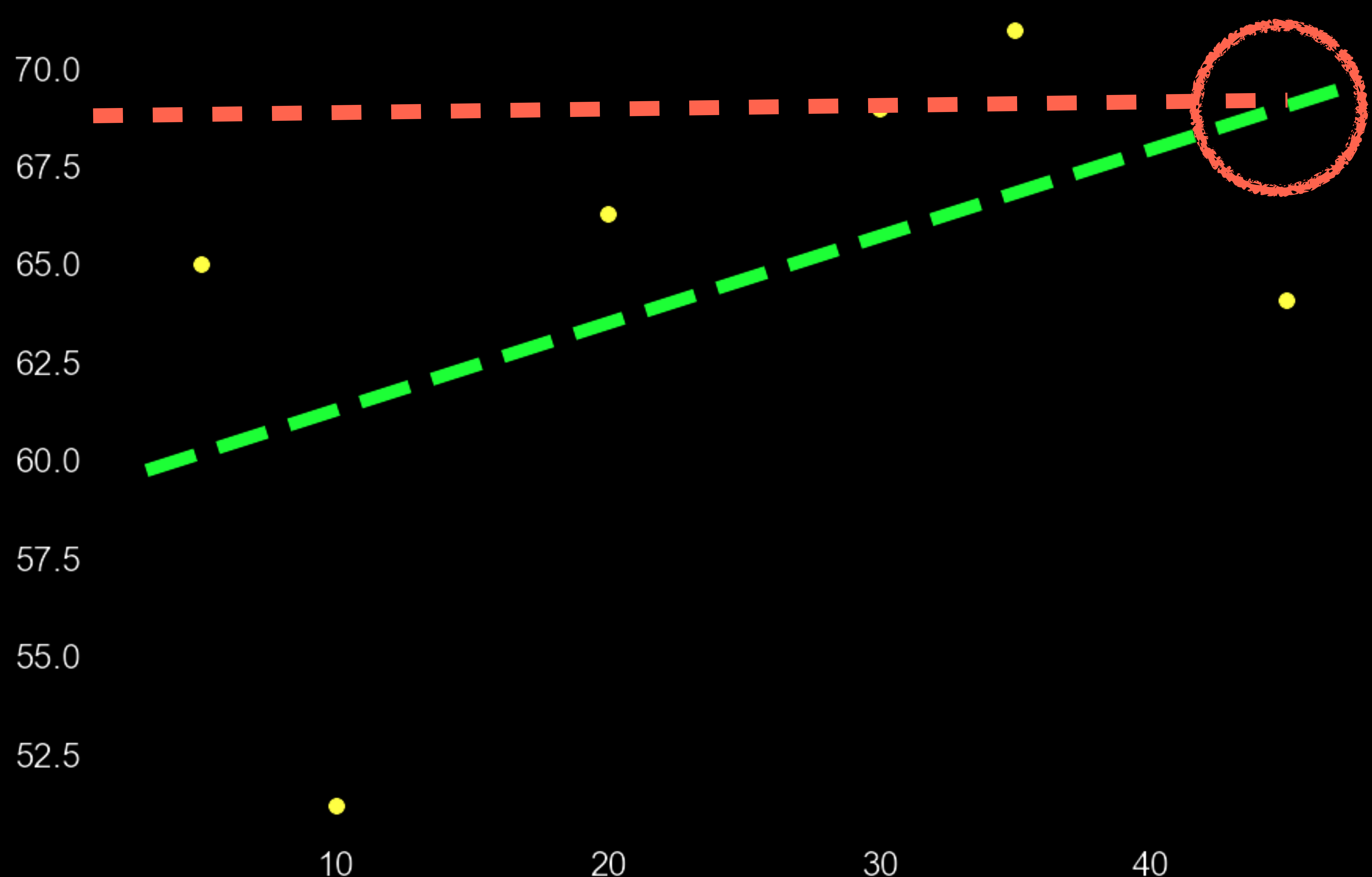
Ожидаемые баллы

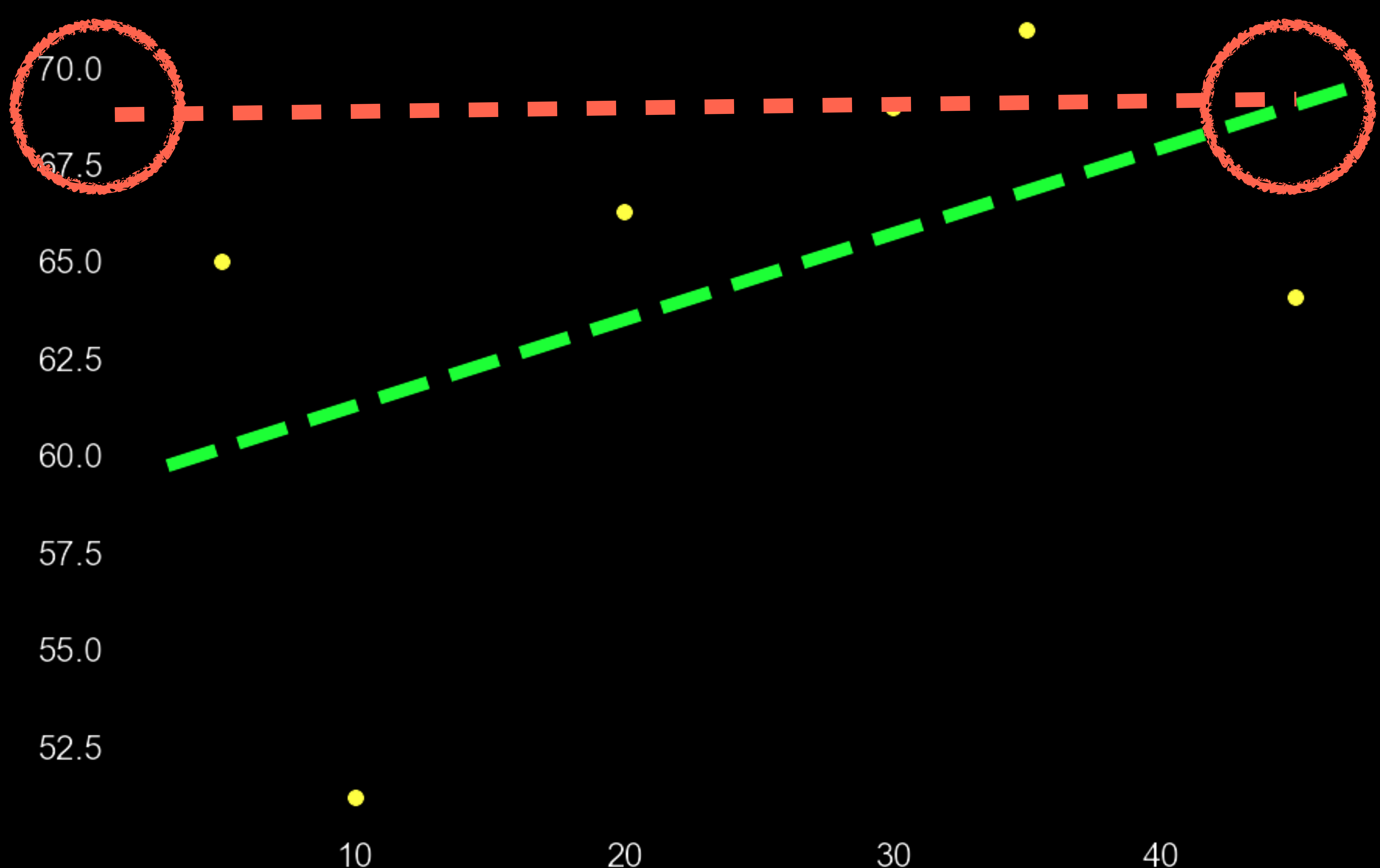
Результаты прошлых лет

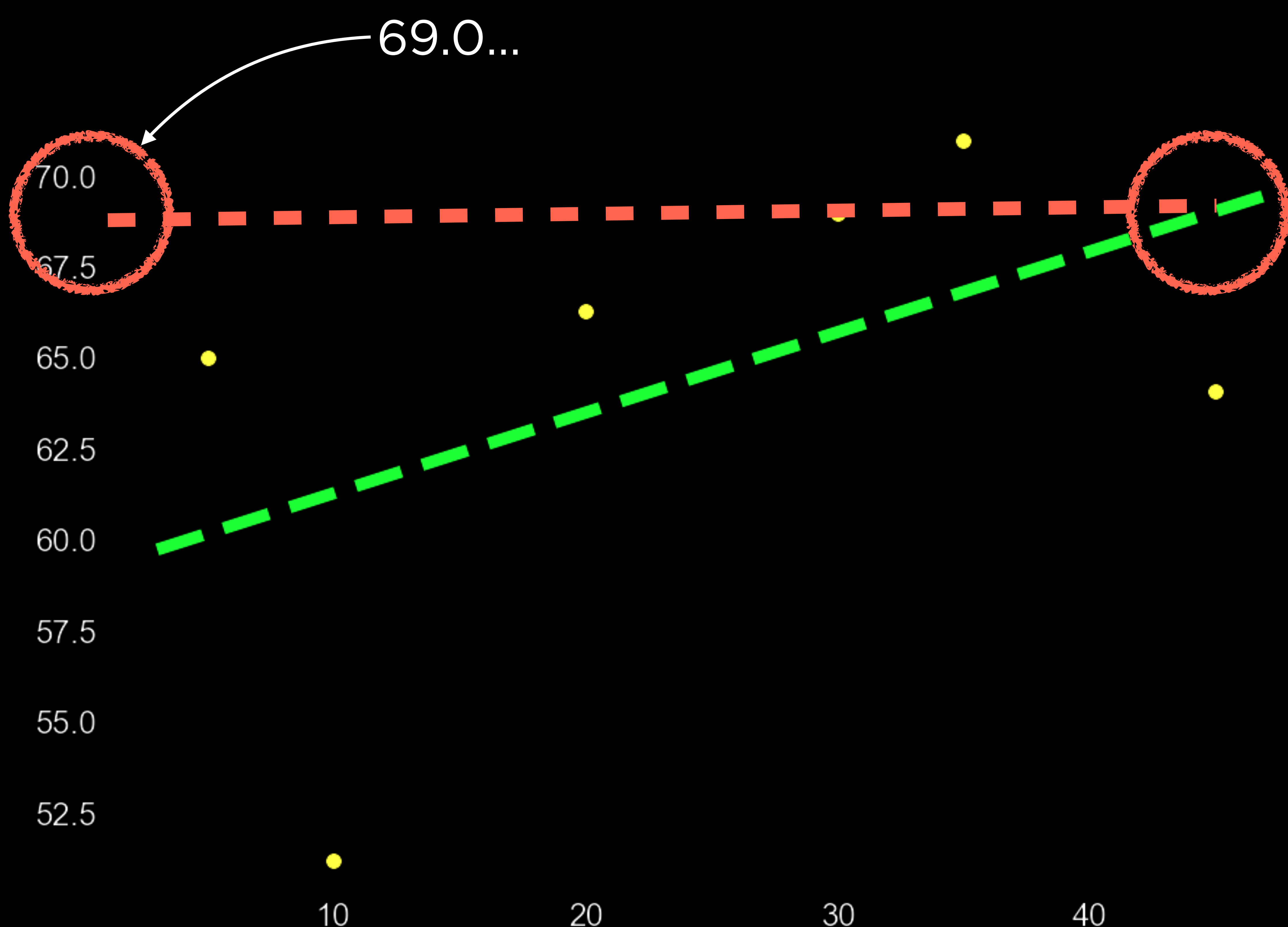
Количество мест

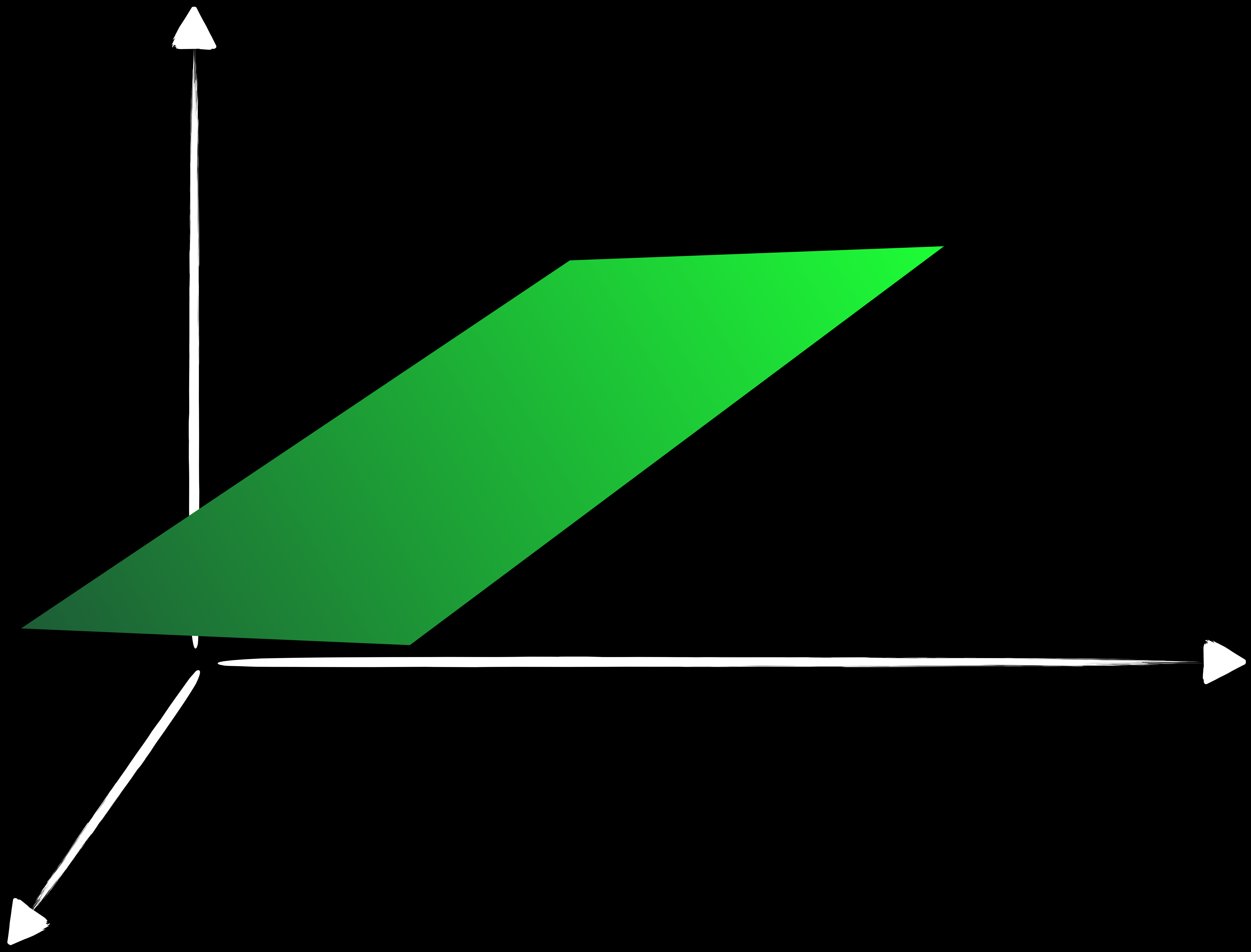

70.0
67.5
65.0
62.5
60.0
57.5
55.0
52.5
10
20
30
40

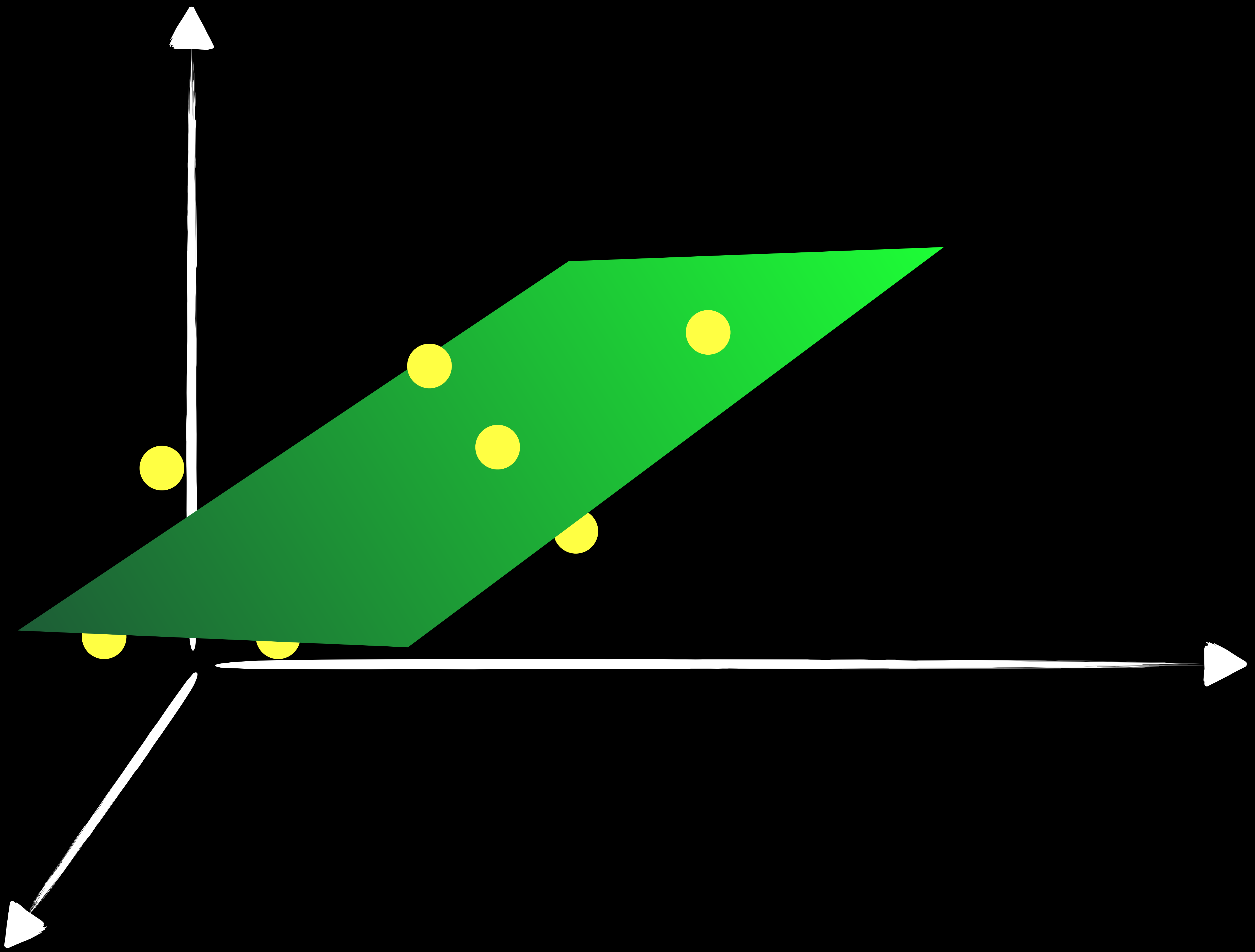

70.0
67.5
65.0
62.5
60.0
57.5
55.0
52.5
10
20
30
40


70.0
67.5
65.0
62.5
60.0
57.5
55.0
52.5
10
20
30
40


70.0
67.5
65.0
62.5
60.0
57.5
55.0
52.5
10
20
30
40


69.0...
70.0
67.5
65.0
62.5
60.0
57.5
55.0
52.5
10
20
30
40





Провести плоскость так,

Провести плоскость так,  чтобы все точки в пространстве

Провести плоскость так,  чтобы все точки в пространстве
были как можно ближе к плоскости

$$Y = Xa + b$$

$$Y = Xa + b$$

$$\underline{Y} = \underline{X}a + b$$

Хотим найти

$$Y = Xa + b$$

$$a = (a_1, \ldots, a_n)$$

Хотим найти

$$Y = Xa + b$$

$$a = (a_1, \ldots, a_n)$$

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \ldots \\ x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

Хотим найти

$$\underline{Y} = \underline{X}a + b$$

$$a = (a_1, \ldots, a_n)$$

Хотим найти

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \cdots \\ x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

$$\underline{Y} = \underline{X}a + b$$

$$\begin{matrix} a_1x_{1,1} + a_2x_{1,2} + a_3x_{1,3} + \ldots + a_nx_{1,n} + b \\ a_1x_{2,1} + a_2x_{2,2} + a_3x_{2,3} + \ldots + a_nx_{2,n} + b \\ \cdots \\ a_1x_{l,1} + a_2x_{l,2} + a_3x_{l,3} + \ldots + a_nx_{l,n} + b \end{matrix}$$

$$a = (a_1, \ldots, a_n)$$

Хотим найти

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \ldots \\ x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

$$\underline{Y} = \underline{X}a + b$$

$$\begin{matrix} a_1 x_{1,1} + a_2 x_{1,2} + a_3 x_{1,3} + \ldots + a_n x_{1,n} + b \\ a_1 x_{2,1} + a_2 x_{2,2} + a_3 x_{2,3} + \ldots + a_n x_{2,n} + b \\ \ldots \\ a_1 x_{l,1} + a_2 x_{l,2} + a_3 x_{l,3} + \ldots + a_n x_{l,n} + b \end{matrix} \longrightarrow$$

$$a = (a_1, \dots, a_n)$$

Хотим найти

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \dots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \dots x_{2,n} \\ \dots \\ x_{l,1}, x_{l,2}, x_{l,3}, \dots x_{l,n} \end{matrix}$$

$$\underline{Y} = \underline{X}a + b$$

$$\begin{matrix} a_1x_{1,1} + a_2x_{1,2} + a_3x_{1,3} + \dots + a_nx_{1,n} + b \\ a_1x_{2,1} + a_2x_{2,2} + a_3x_{2,3} + \dots + a_nx_{2,n} + b \\ \dots \\ a_1x_{l,1} + a_2x_{l,2} + a_3x_{l,3} + \dots + a_nx_{l,n} + b \end{matrix} \longrightarrow \begin{matrix} +1 * b \\ \cdots +1 * b \\ \dots \\ +1 * b \end{matrix}$$

$$a = (a_1, \ldots, a_n)$$

Хотим найти

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \cdots \\ x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

$$Y = X'w$$

$$\begin{matrix} a_1x_{1,1} + a_2x_{1,2} + a_3x_{1,3} + \ldots + a_nx_{1,n} + b \\ a_1x_{2,1} + a_2x_{2,2} + a_3x_{2,3} + \ldots + a_nx_{2,n} + b \\ \cdots \\ a_1x_{l,1} + a_2x_{l,2} + a_3x_{l,3} + \ldots + a_nx_{l,n} + b \end{matrix} \longrightarrow \begin{matrix} +1 * b \\ \cdots +1 * b \\ \cdots \\ +1 * b \end{matrix}$$

$$w = (b, a_1, \ldots, a_n)$$

$$a = (a_1, \ldots, a_n)$$

Хотим найти

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \cdots \\ x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

$$Y = X'w$$

$$\begin{matrix} a_1x_{1,1} + a_2x_{1,2} + a_3x_{1,3} + \ldots + a_nx_{1,n} + b \\ a_1x_{2,1} + a_2x_{2,2} + a_3x_{2,3} + \ldots + a_nx_{2,n} + b \\ \cdots \\ a_1x_{l,1} + a_2x_{l,2} + a_3x_{l,3} + \ldots + a_nx_{l,n} + b \end{matrix} \longrightarrow \begin{matrix} +1 * b \\ \cdots +1 * b \\ \cdots \\ +1 * b \end{matrix}$$

$$w = (b, a_1, \ldots, a_n)$$

$$a = (a_1, \ldots, a_n)$$

Хотим найти

$$X = \begin{matrix} x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \cdots \\ x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

$$X = \begin{matrix} 1, x_{1,1}, x_{1,2}, x_{1,3}, \ldots x_{1,n} \\ 1, x_{2,1}, x_{2,2}, x_{2,3}, \ldots x_{2,n} \\ \cdots \\ 1, x_{l,1}, x_{l,2}, x_{l,3}, \ldots x_{l,n} \end{matrix}$$

$$Y = X'w$$

$$\begin{matrix} a_1 x_{1,1} + a_2 x_{1,2} + a_3 x_{1,3} + \ldots + a_n x_{1,n} + b \\ a_1 x_{2,1} + a_2 x_{2,2} + a_3 x_{2,3} + \ldots + a_n x_{2,n} + b \\ \cdots \\ a_1 x_{l,1} + a_2 x_{l,2} + a_3 x_{l,3} + \ldots + a_n x_{l,n} + b \end{matrix} \longrightarrow \begin{matrix} +1 * b \\ \cdots +1 * b \\ \cdots \\ +1 * b \end{matrix}$$

$$Y = X'w$$

$$Y = X'w$$

$$w = (X')^{-1}Y$$

$$Y = X'w$$

$$dim(X) = [l, n], l \neq n$$

# Метод наименьших квадратов

# Формулировка МНК

$$(Xw - y)^T (Xw - y) \to \min_w$$

$$Xw = Y$$

$$Xw = Y$$
$$X^T X w = X^T Y$$

$$Xw = Y$$
$$X^T X w = X^T Y$$
$$w = (X^T X)^{-1} X^T Y$$

# Проблемы

# Проблемы

- Имеется операция взятия обратной матрицы

# Проблемы

$O(n^3)$ операций

- Имеется операция взятия обратной матрицы

# Проблемы

$O(n^3)$ операций

- Имеется операция взятия обратной матрицы
- Могут быть линейно зависимые вектора

# Проблемы

$O(n^3)$ операций

- Имеется операция взятия обратной матрицы
- Могут быть линейно зависимые вектора

Тогда матрица
необратима

Что мы делаем, когда невозможно аналитически?

Аппроксимируем!

# Градиентный спуск

# Градиентный спуск

$$(Xw - y)^T(Xw - y) \to \min_w$$

# Градиентный спуск

$$(Xw - y)^T(Xw - y) \to \min_w$$

$$\frac{1}{l}||Xw - y||^2 \to \min_w$$

# Градиентный спуск

$Q(w, X)$

$$(Xw - y)^T(Xw - y) \to \min_w$$

$$\frac{1}{l}||Xw - y||^2 \to \min_w$$

# Градиентный спуск

# Градиентный спуск

$$w^0 = 0$$

# Градиентный спуск

$$w^0 = 0$$

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

# Градиентный спуск

$$w^0 = 0$$

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

$$||w^t - w^{t-1}|| < \epsilon$$

# Градиентный спуск

Шаг
градиентного
спуска

$$w^0 = 0$$

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

$$||w^t - w^{t-1}|| < \epsilon$$

# Градиентный спуск

Шаг градиентного спуска

$$w^0 = 0$$

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

Градиент от функции потерь по $w$

$$||w^t - w^{t-1}|| < \epsilon$$

# Градиентный спуск

Веса на предыдущем шаге

$$w^0 = 0$$

Шаг градиентного спуска

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

Градиент от функции потерь по $w$

$$||w^t - w^{t-1}|| < \epsilon$$

# Градиентный спуск

Веса на предыдущем шаге

$$w^0 = 0$$

Шаг градиентного спуска

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

Градиент от функции потерь по $w$

$$||w^t - w^{t-1}|| < \epsilon$$

Очень маленькое значение

# Комментарий насчет шага

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

# Комментарий насчет шага

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

# Комментарий насчет шага

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

$$\eta_t = \frac{k}{t}$$

# Комментарий насчет шага

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

Константа
(сами выбираем)

$$\eta_t = \frac{k}{t}$$

# Комментарий насчет шага

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, X)$$

Константа
(сами выбираем)

$$\eta_t = \frac{k}{t}$$

Номер итерации

# Как посчитать градиент?

# Like baby steps

$$f(x) = w_0 + xw_1$$

$$Q(w_0, w_1, X) = \frac{1}{l}\sum_{i=1}^{l}(w_1 x_i + w_0 - y_i)^2$$

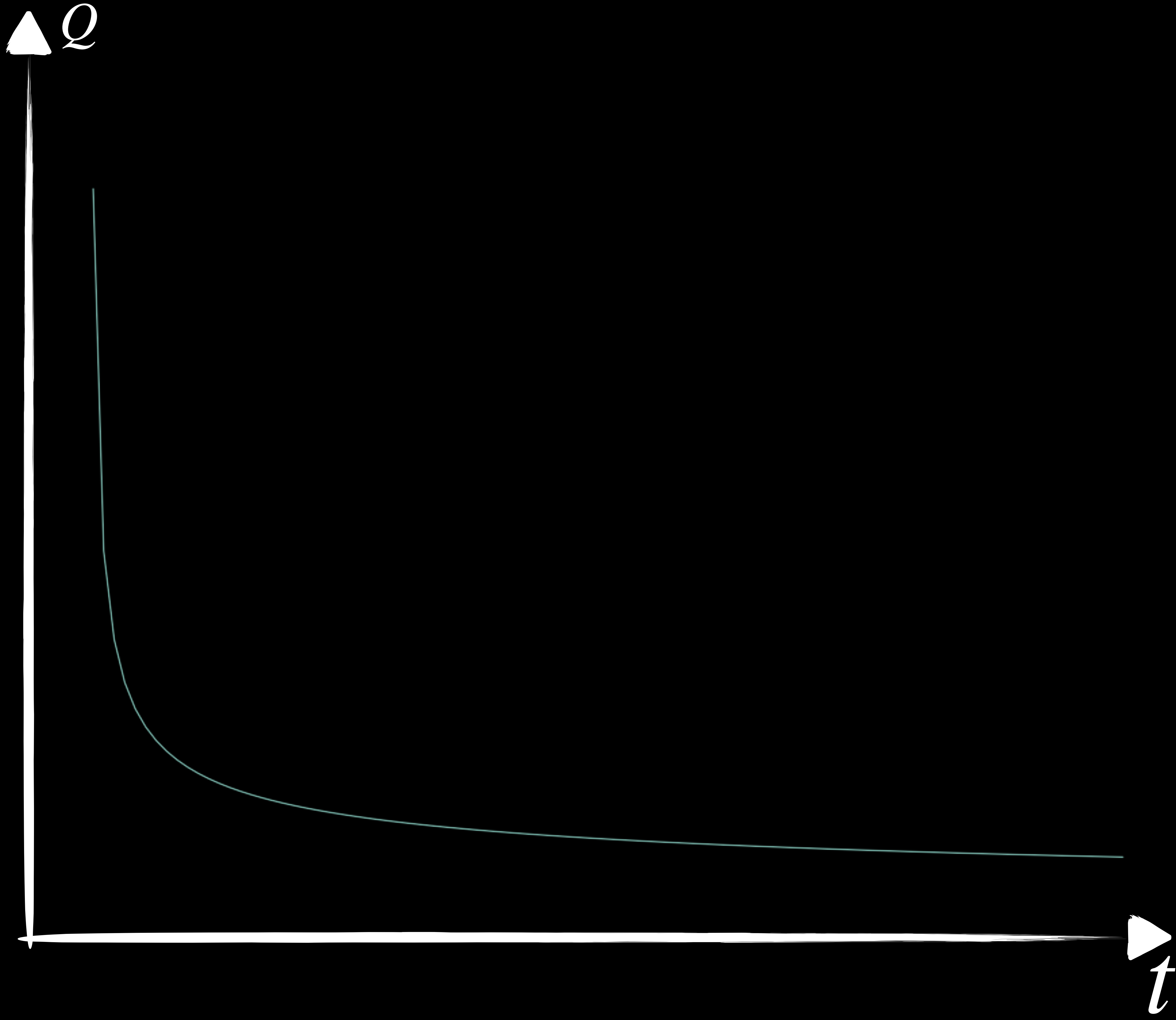
$$\frac{\partial Q}{\partial w_1} = \frac{2}{l}\sum_{i=1}^{l}(w_1 x_i + w_0 - y_i)x_i$$

$$\frac{\partial Q}{\partial w_0} = \frac{2}{l}\sum_{i=1}^{l}(w_1 x_i + w_0 - y_i)$$

# Обобщим

$$Q(w, X) = \frac{1}{l} ||Xw - y||^2 \to \min_w$$

$$\nabla Q(w, X) = \frac{2}{l} X^T (Xw - y)$$


Q
t

# Проблемы

- Он может долго считаться при большой выборке

# Проблемы

- Он может долго считаться при большой выборке

$$\nabla Q(w, X) = \frac{2}{l} X^T (Xw - y)$$

# Проблемы

- Он может долго считаться при большой выборке

$$\nabla Q(w, X) = \frac{2}{l} X^T (Xw - y)$$

Примеров может
быть очень много

# Стохастический градиентный спуск

# Стохастический градиентный спуск

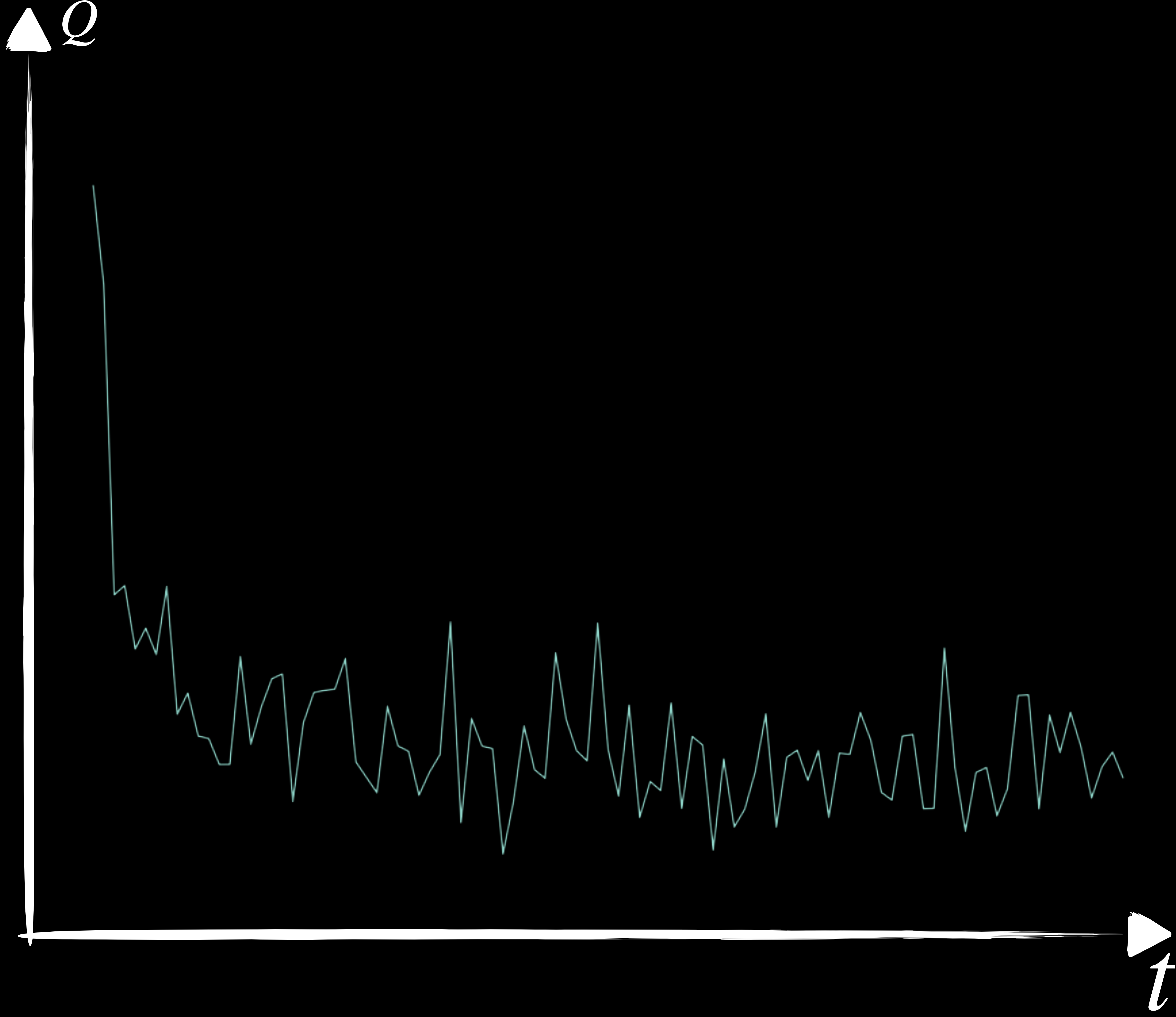
$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, \{x_i\})$$

# Стохастический градиентный спуск

Кусок выборки

$$w^t = w^{t-1} - \eta_t \nabla Q(w^{t-1}, \{x_i\})$$

Берем
случайные
куски выборки



Q
t

# Хорошие вещи

- Легко дообучить, если пришли новые данные;
- Достаточно быстро работает.

# Хорошие вещи

Изначально
учимся по кускам

- Легко дообучить, если пришли новые данные;
- Достаточно быстро работает.

# Классификация

# Бинарная классификация

# Постановка задачи

$Y \in \mathbb{R}$ Регрессия

# Постановка задачи

~~$Y \in \mathbb{R}$ Регрессия~~

# Постановка задачи

~~$Y \in \mathbb{R}$ Регрессия~~

$$Y \in \{-1, 1\}$$

Бинарная классификация

$$Y \in \{0, 1\}$$

# Пример

# Пример

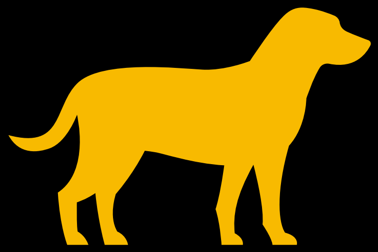
1

0

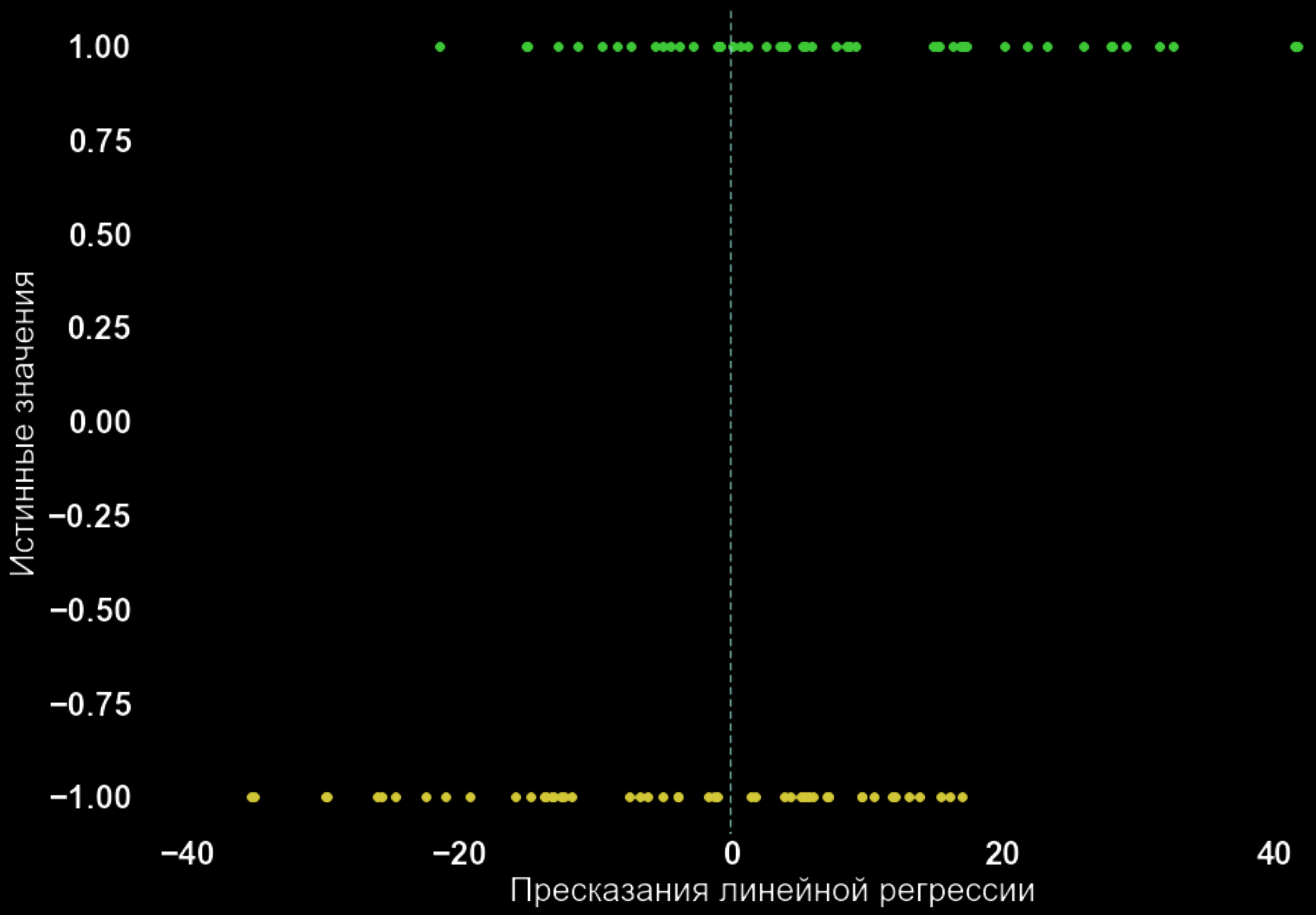
# Пример

1

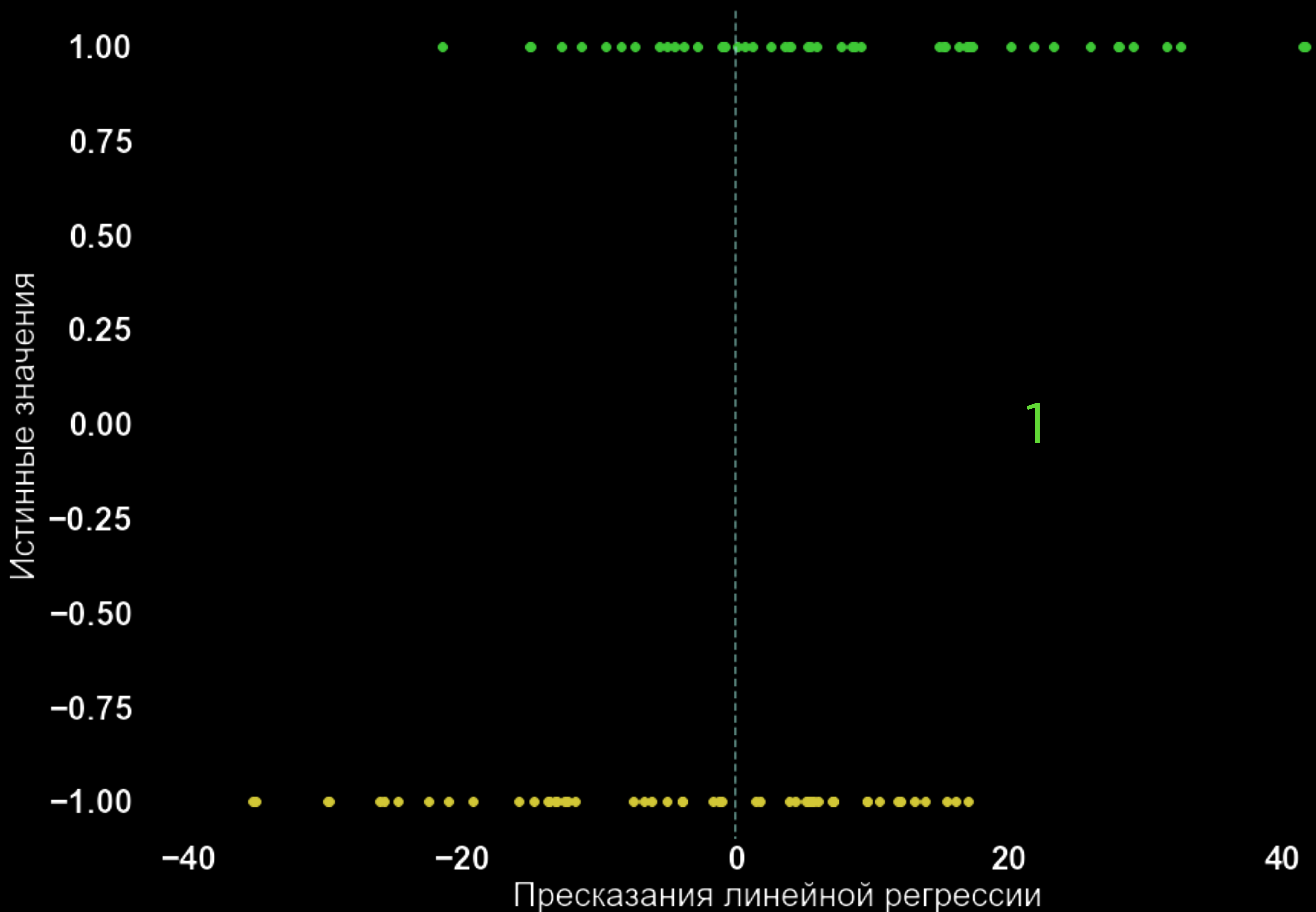
0



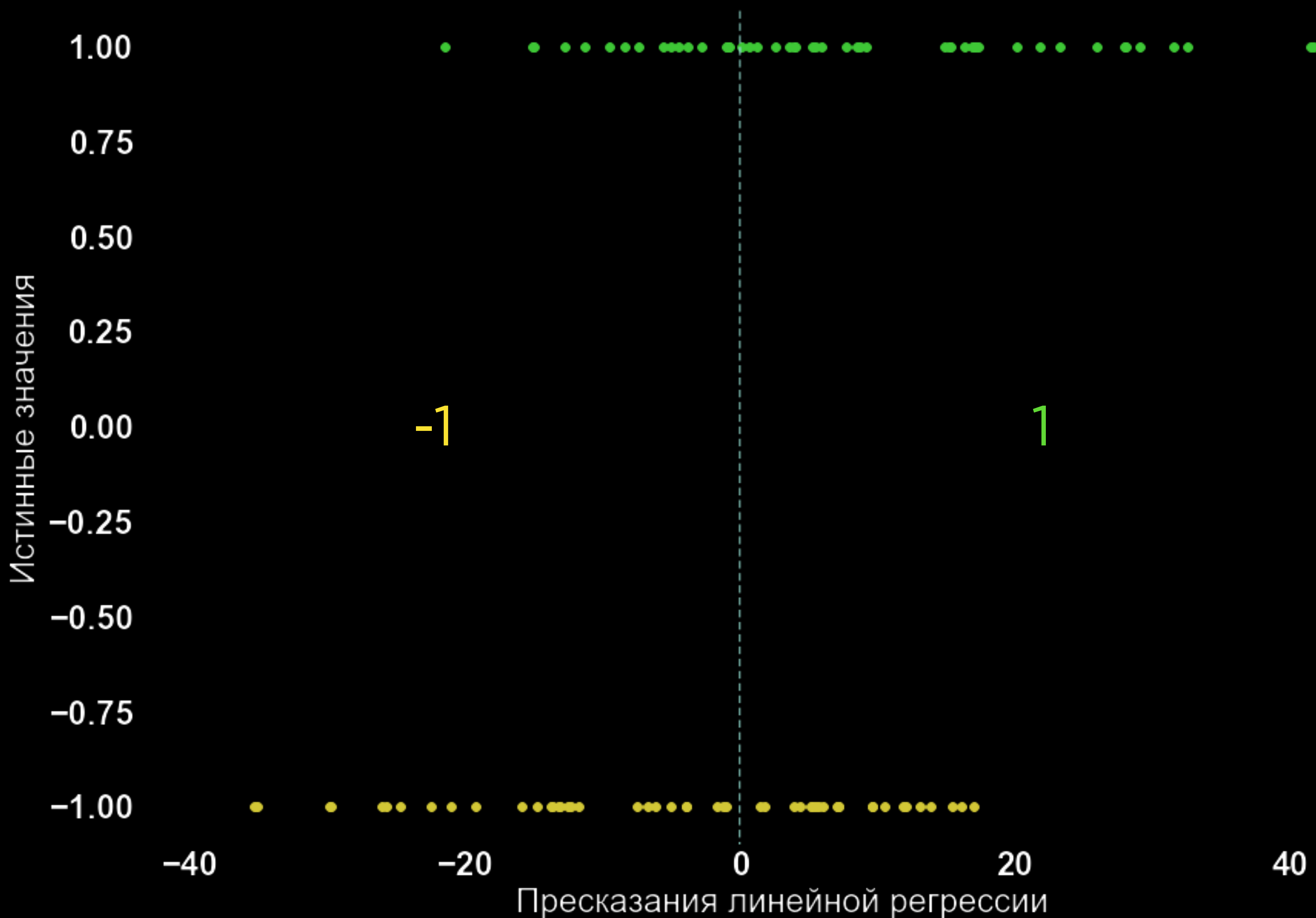
# Регрессия для классификации



# Регрессия для классификации



# Регрессия для классификации



# От регрессии к классификации

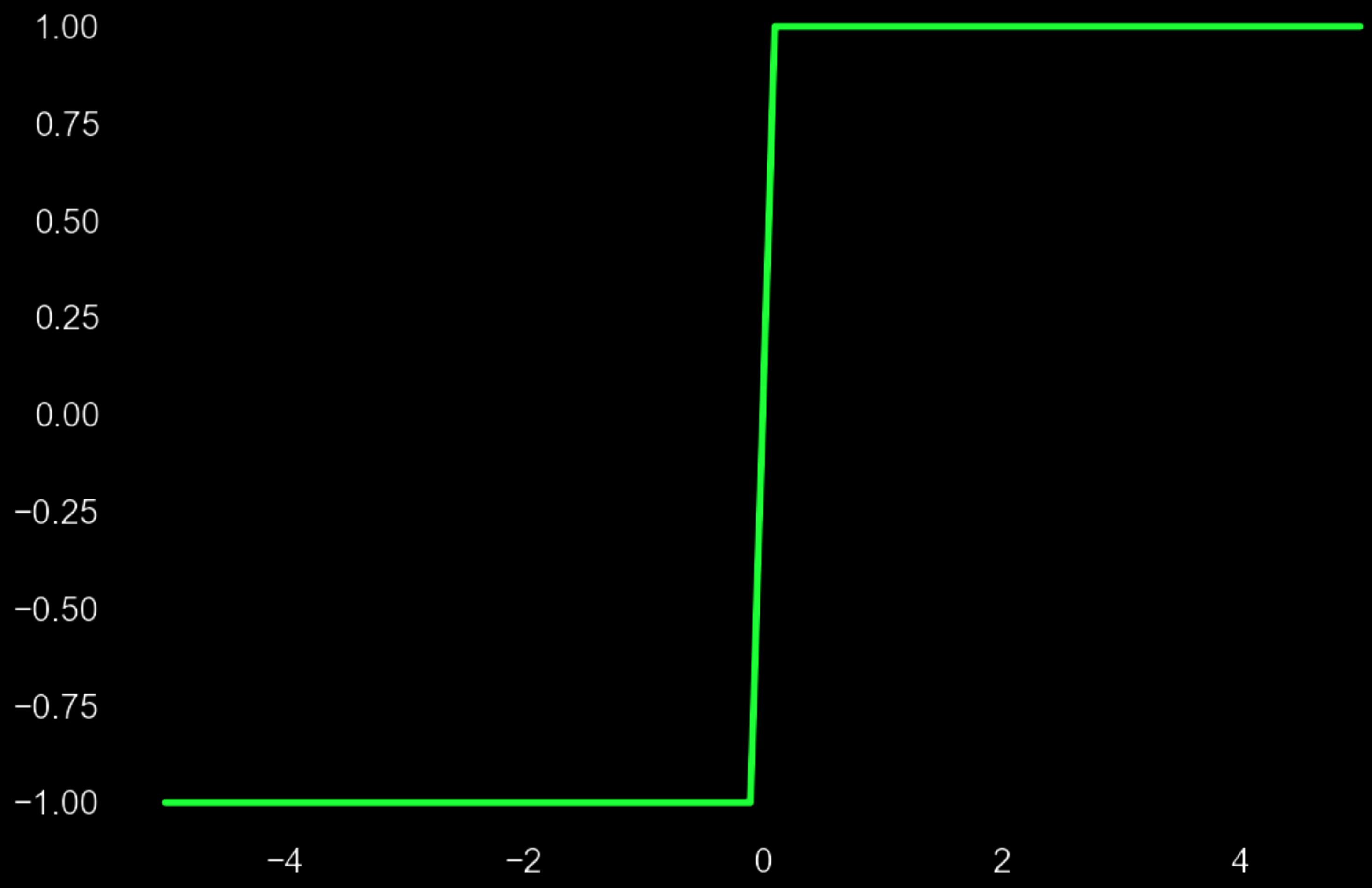
$$Y = Xw + b$$

# От регрессии к классификации

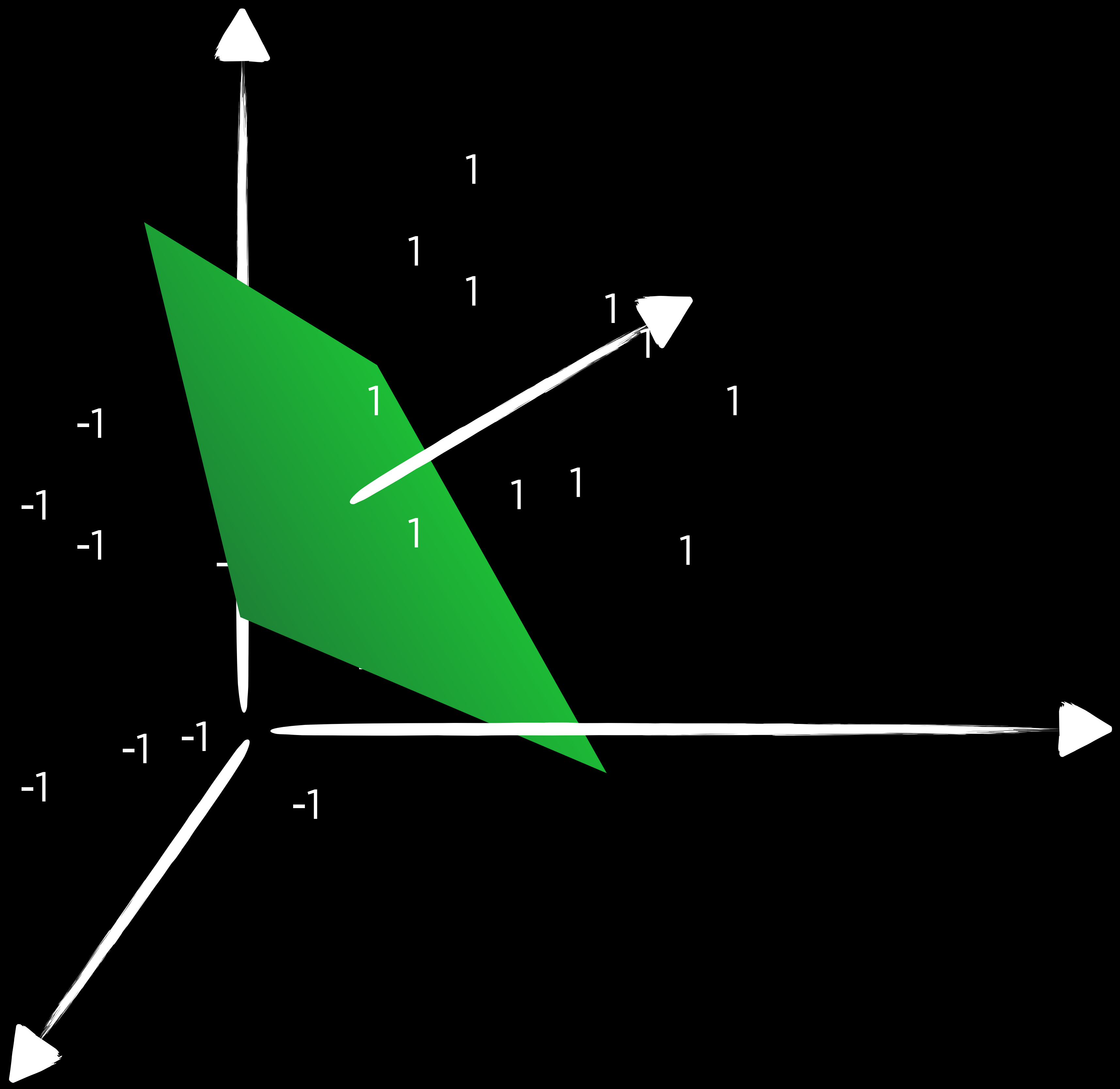
$$Y = sign\left(Xw + b\right)$$

# От регрессии к классификации

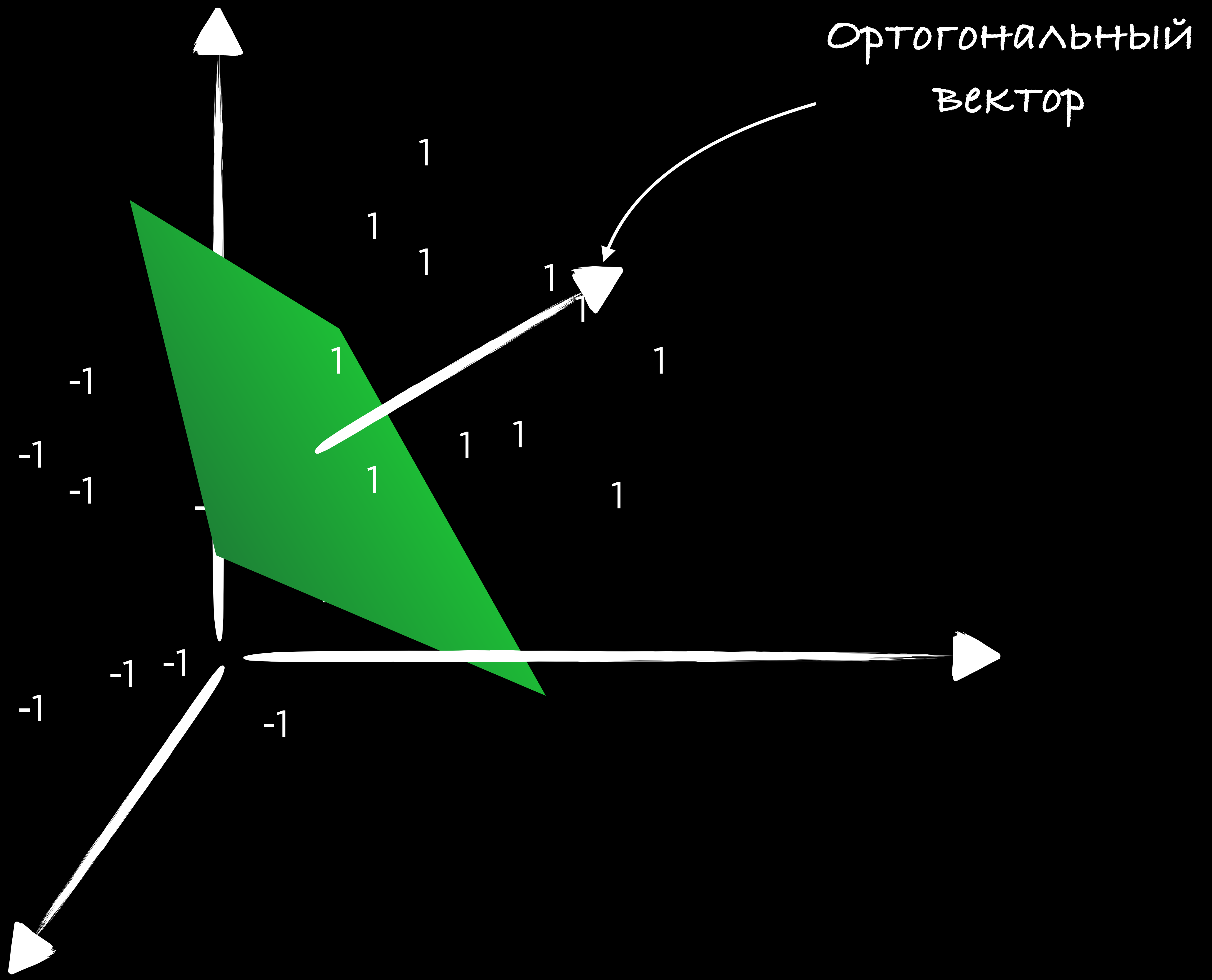
$$sign(x) = \begin{cases} 1 \ if \ x > 0 \\ 0 \ if \ x == 0 \\ -1 \ if \ x < 0 \end{cases} \quad Y = sign(Xw + b)$$



# Почему?



# Почему?



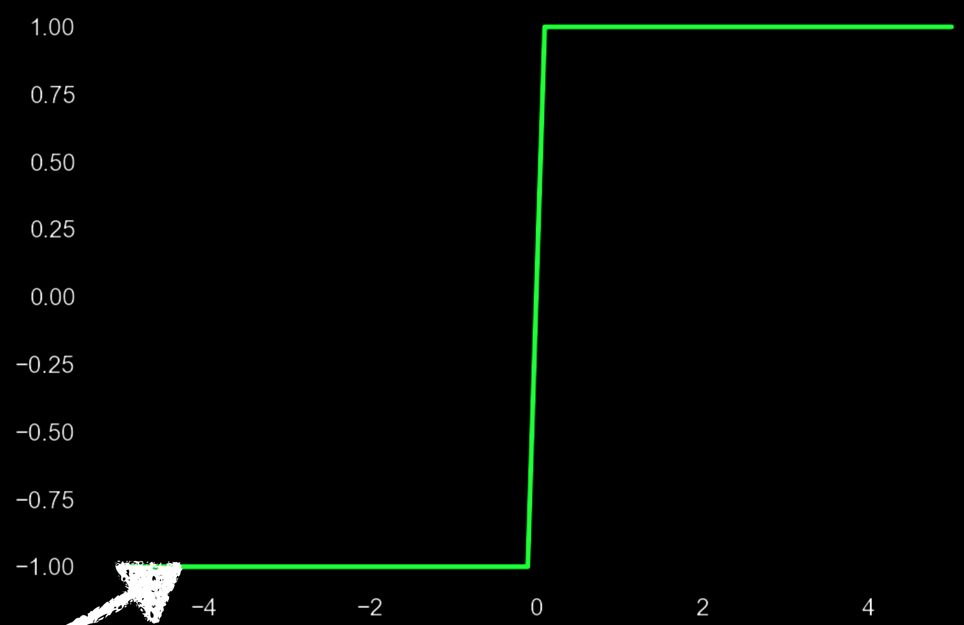
# Но

- Хотим использовать методы оптимизации, которые выше

# Но

- Хотим использовать методы оптимизации, которые выше

Нелинейная штука

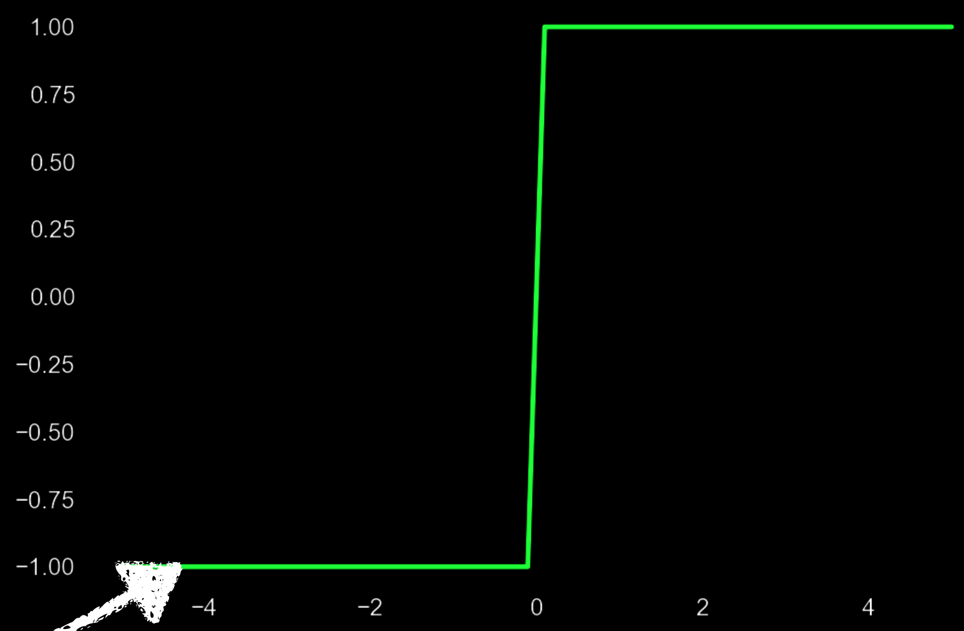


# Но

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука



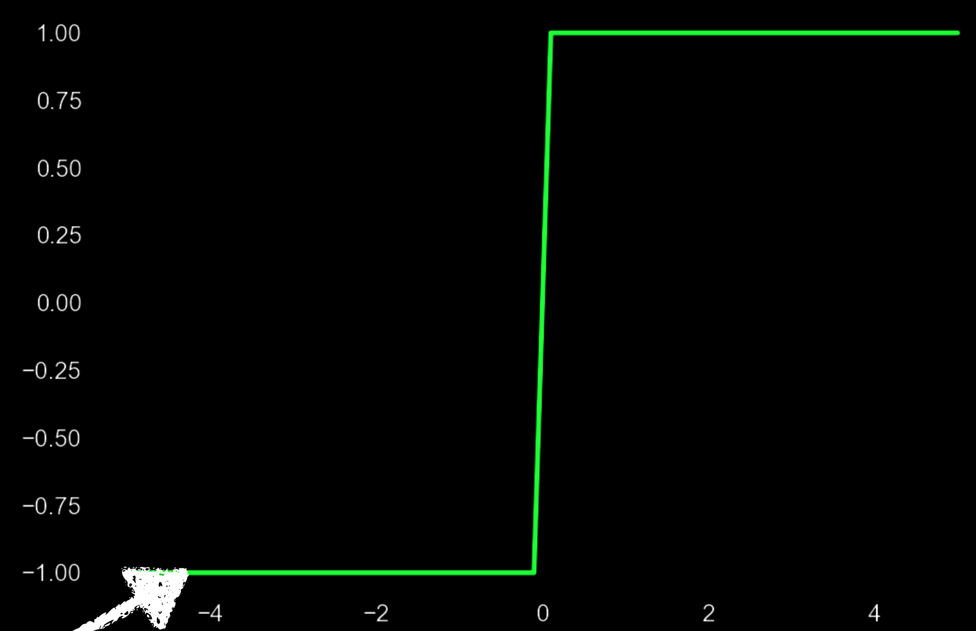
# Но

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука



# Но

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

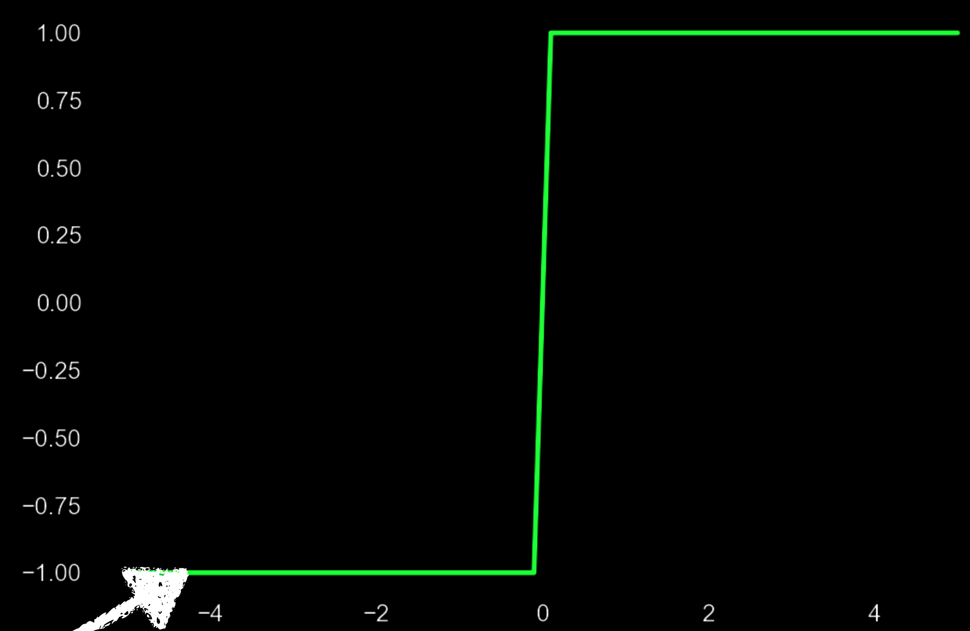
Раньше

$Xw + b$

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука



# Но

Теперь

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

$sign(Xw + b)$

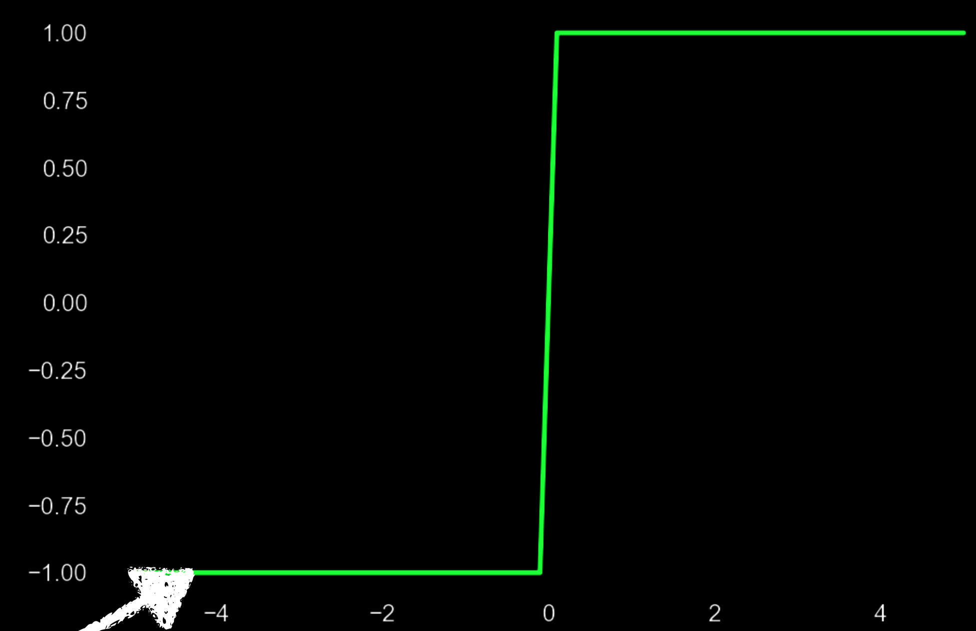
Раньше

$Xw + b$

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука



# Но

Теперь

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

$sign(Xw + b)$

Раньше

$Xw + b$

$\nabla Q$

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука

# Но

$$Q = \frac{1}{l}||sign(Xw+b) - y||^2$$

Теперь

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

$sign(Xw+b)$

Раньше

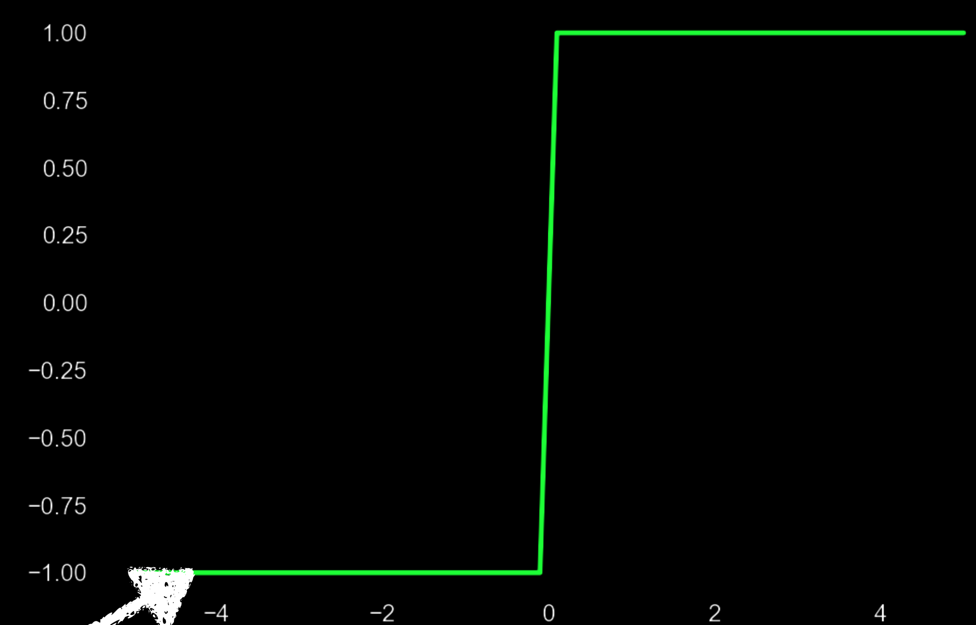
$Xw + b$

$\nabla Q$

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука



# Но

$$Q = \frac{1}{l} || sign(Xw + b) - y||^2$$

Теперь

$$Y = f(X_1, \ldots, X_l) + \epsilon$$

$sign(Xw + b)$

Раньше

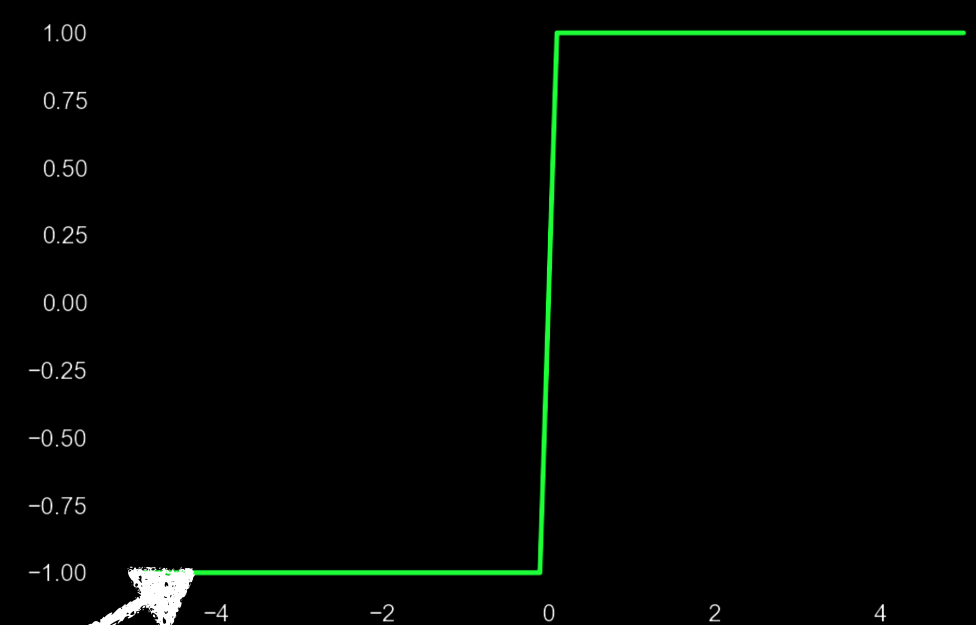
$Xw + b$

$\nabla Q$

- Хотим использовать методы оптимизации, которые выше

А нам нужны производные

Нелинейная штука



# Логистическая регрессия

# Логистическая регрессия

Logistic for logits

Logits

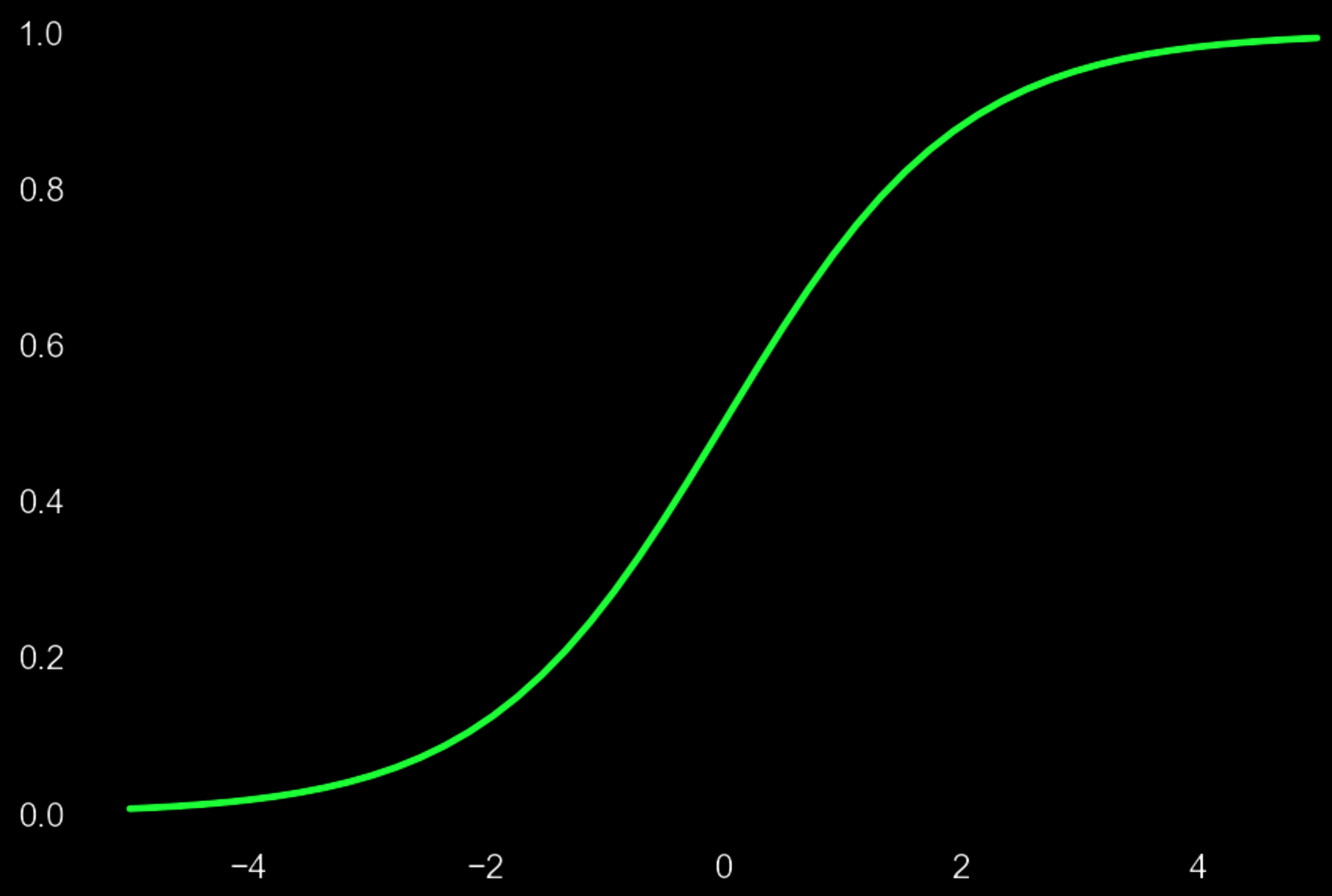
$$p = sigmoid(Xw + b)$$

# Что за сигмойда?

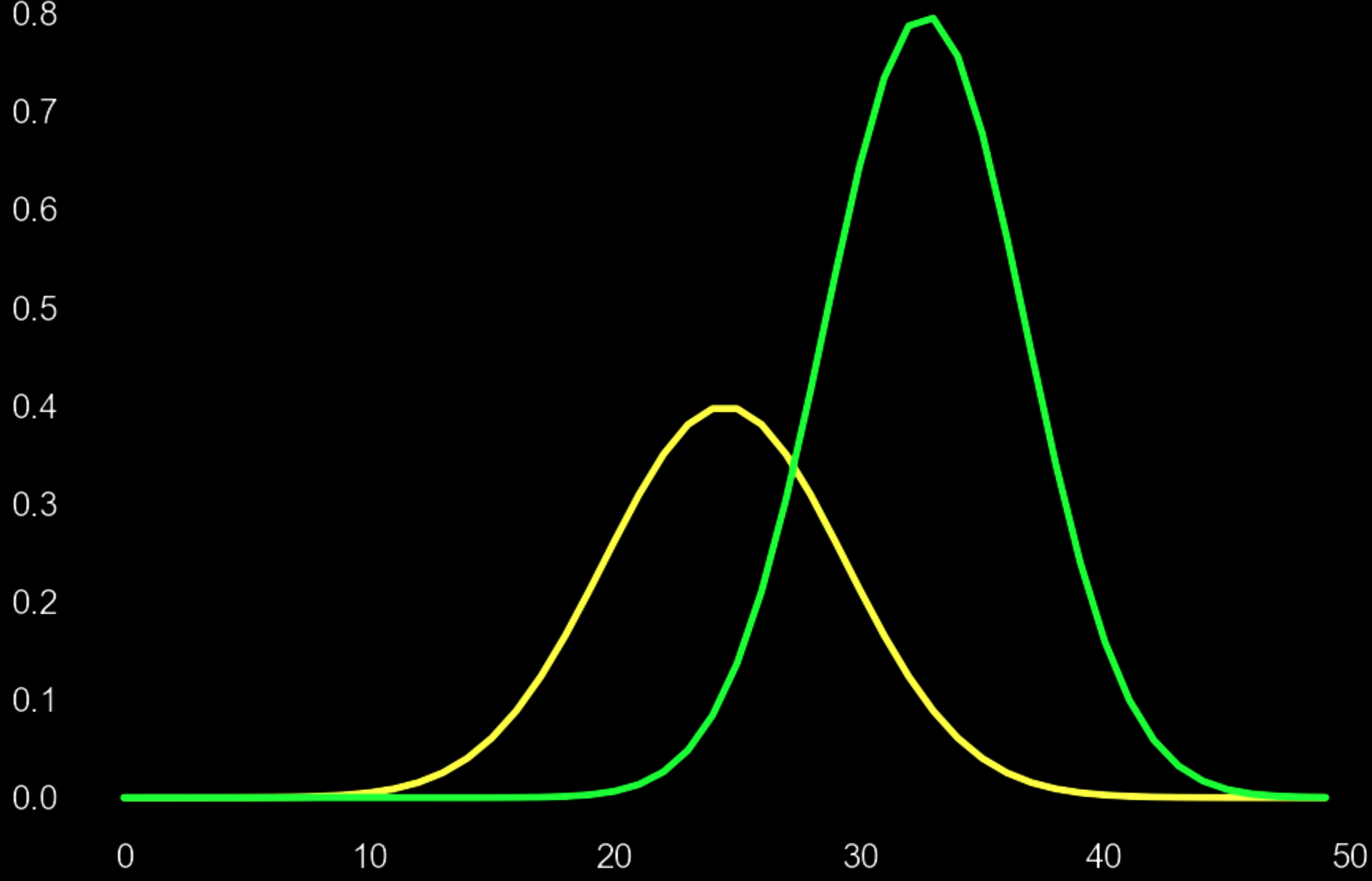
$$sigmoid(x) = \frac{1}{1 + e^{-x}}$$

# Что за сигмойда?

$$sigmoid(x) = \frac{1}{1 + e^{-x}}$$



# Функция потерь


0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
10
20
30
40
50

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i)log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

Велико если неправильно

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$y_i = 1$ $\qquad$ $y_i = 0$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

Велико если неправильно

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$$y_i = 0$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(1 - p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

Велико если неправильно

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

Велико если неправильно

$$y_i = 0$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(1 - p_i)$$

$p_i$ Как можно ближе к 0

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

Велико если неправильно

$$y_i = 0$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(1 - p_i)$$

$p_i$ Как можно ближе к 0

Мало если правильно

# Cross entropy

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i) log(1 - p_i))$$

$$y_i = 1$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(p_i)$$

$p_i$ Как можно ближе к 1

Мало если правильно

Велико если неправильно

$$y_i = 0$$

$$J(w) = -\frac{1}{N}\sum_{i=1}^{l} log(1 - p_i)$$

$p_i$ Как можно ближе к 0

Мало если правильно

Велико если неправильно

# Градиент

# Функции ошибки

# Функции ошибки

$$Q(w, X) = \frac{1}{l}||Xw - y||^2 \to \min_w$$

# Функции ошибки

$$Q(w, X) = \frac{1}{l}||Xw - y||^2 \to \min_{w}$$

$$J(w, X) = \frac{1}{l}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i)log(1 - p_i)) \to \min_{w}$$

# Функции ошибки

Раньше

$$Q(w, X) = \frac{1}{l}||Xw - y||^2 \to \min_w$$

$$J(w, X) = \frac{1}{l}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i)log(1 - p_i)) \to \min_w$$

# Функции ошибки

Раньше

$$Q(w, X) = \frac{1}{l}||Xw - y||^2 \to \min_w$$

$$J(w, X) = \frac{1}{l}\sum_{i=1}^{l}(y_i log(p_i) + (1 - y_i)log(1 - p_i)) \to \min_w$$

Теперь

# Функции ошибки

Раньше

$$Q(w, X) = \frac{1}{l} || Xw - y ||^2 \to \min_w$$

$$J(w, X) = \frac{1}{l} \sum_{i=1}^{l} (y_i log(p_i) + (1 - y_i) log(1 - p_i)) \to \min_w$$

Теперь

Тут W

# Объяснение, как считать

Как считается производная cross entropy

# Как теперь обновляем веса

$$dw = \frac{1}{l} X^T (A - y)$$

$$db = \frac{1}{l} \sum (A - y)$$

$$w_{new} = w - \eta_t dw$$

$$b_{new} = b - \eta_t db$$

# Литература

- Конспект лекций по линейным моделям
- Полезные видосы по логистической регрессии
- Как считается производная cross entropy

# Контакты

- Телеграмм для вопросов по заданиям: @kuparez
- Телеграмм бот для сдачи заданий: @dsmatmech2018_bot

# Задания

https://github.com/kuparez/data-science-101

# Задания

- Либо реализовать алгоритмы (let’s get hands dirty.ipynb);
  - 10 баллов
- Либо побить baseline решение Титаника (titanic_predictions.ipynb)
  - 10 баллов
- Оба задания: 15 баллов

# Задания

- Мягкий дэдлаин: 17 ноября
  - После него будет даваться половина баллов
- Жесткий дедлаин: 24 ноября
  - После него 0 баллов

# Реализация алгоритмов

- Свои реализации надо отправлять в бот с названием файла “name_surname_ngroup_1.html”
- Ожидается, что Ваши реализации будут вести себя сходно с тем, что написано в ноутбуке
- На вопросы, которые есть в ноутбуке отвечать необязательно, но мы с удовольствие почитаем.

# Титаник

- Требуется побить решение, предложенное нами.
- Надо отправлять решения на Kaggle с тэгом Your Name [mm_ds_course]
- Также Ваши решения ожидаются в телеграмм боте в виде html с названием файла: “name_surname_номер_группы_2.html”
- Более подробные правила расписаны в ноутбуке