

Знакомство с классификацией на примере логистической регрессии

Никита

Сегодня в программе

- Задачи машинного обучения
- Регрессия
- Классификация

Сегодня мы обсудим такие небольшие темы

1. Какие вообще бывают задачи машинного обучения
2. Поставим задачу регрессии и рассмотрим один метод для ее решения
3. Рассмотрим задачу классификации и перейдем от линейной регрессии к классификации.

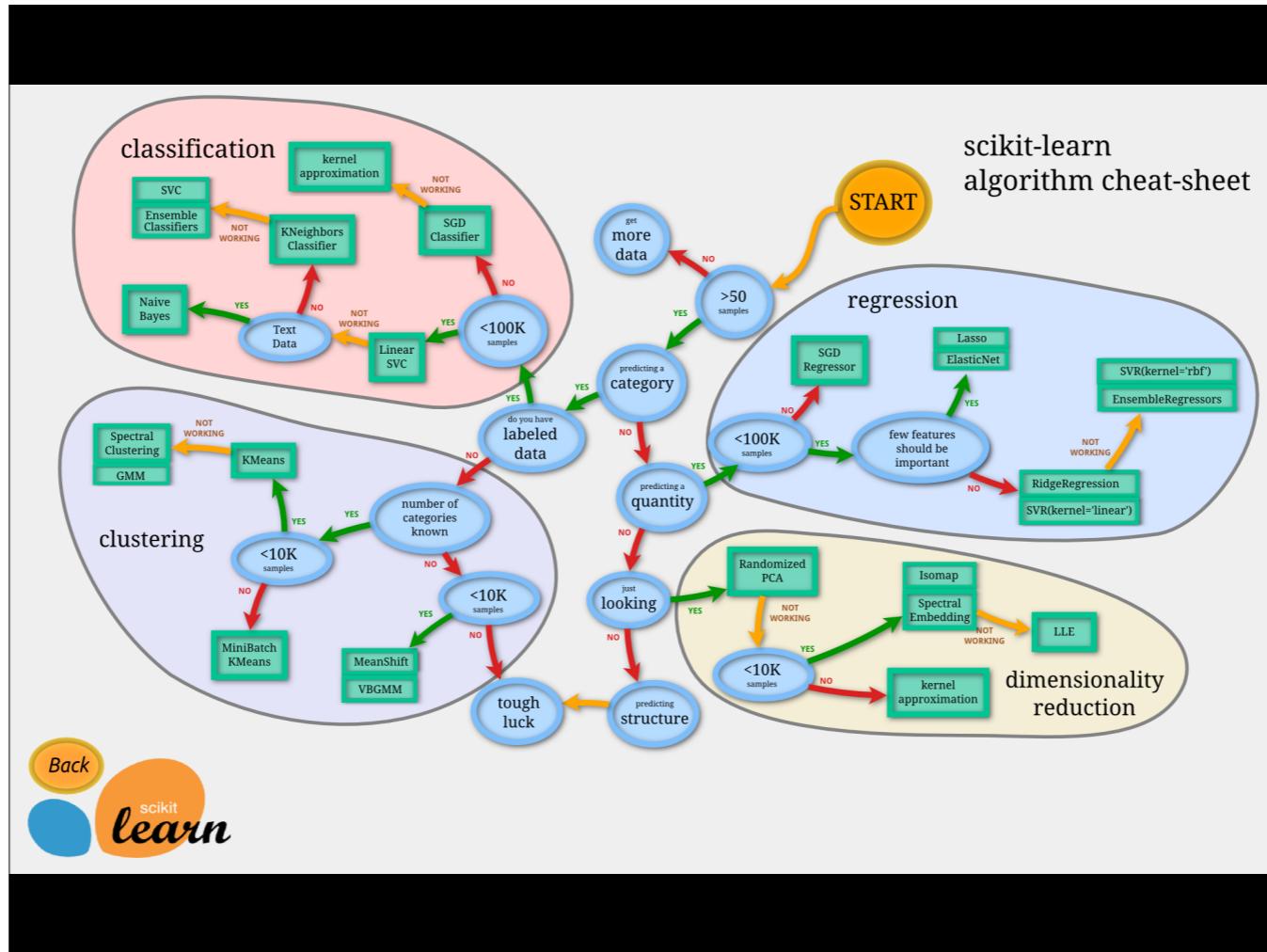
Что в свою очередь перерастает в следующие подпункты:

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?

Но обо всем по порядку

Задачи машинного обучения

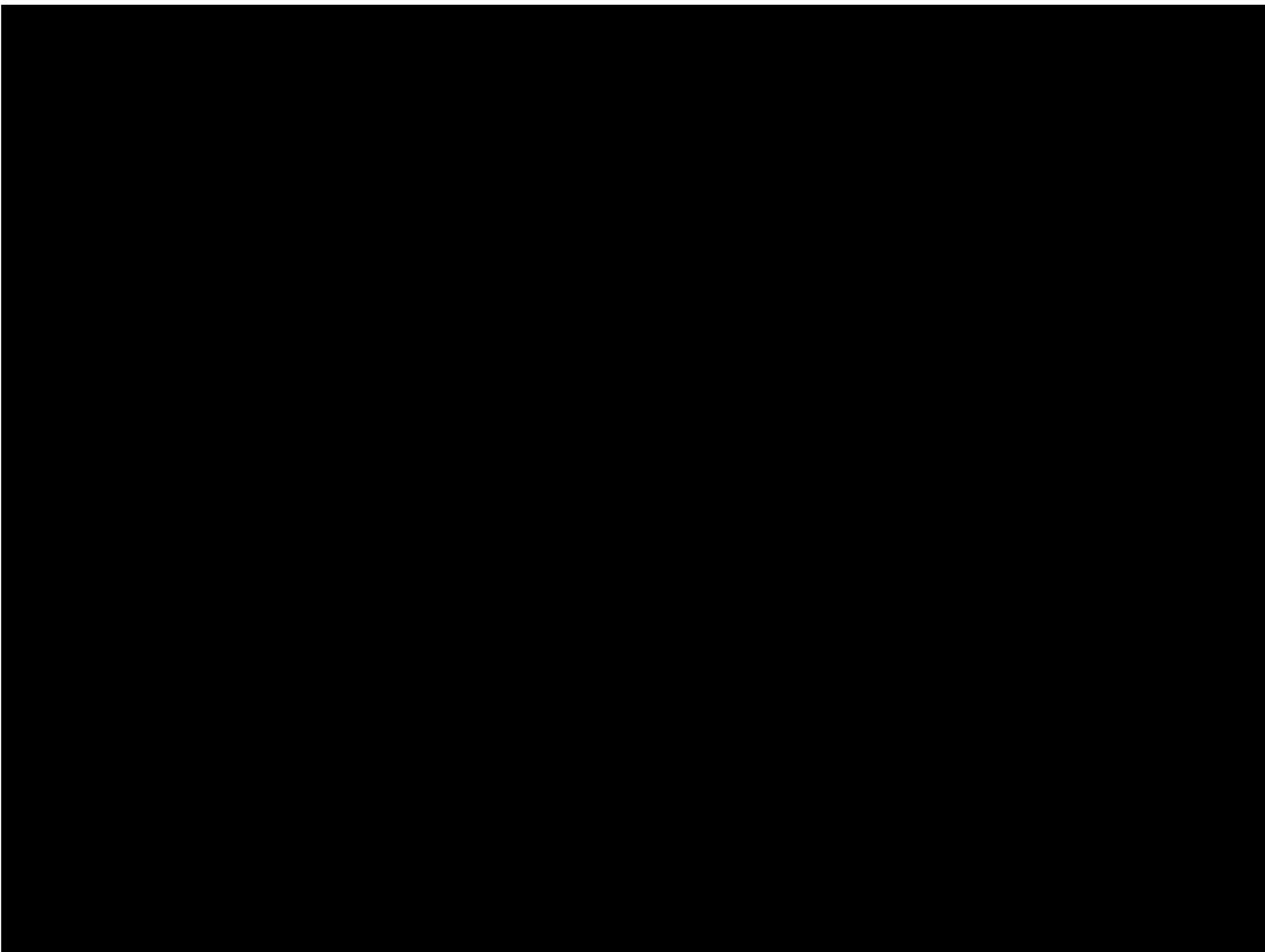


Есть много картинок, которые показывают все возможные отрасли машинного обучения

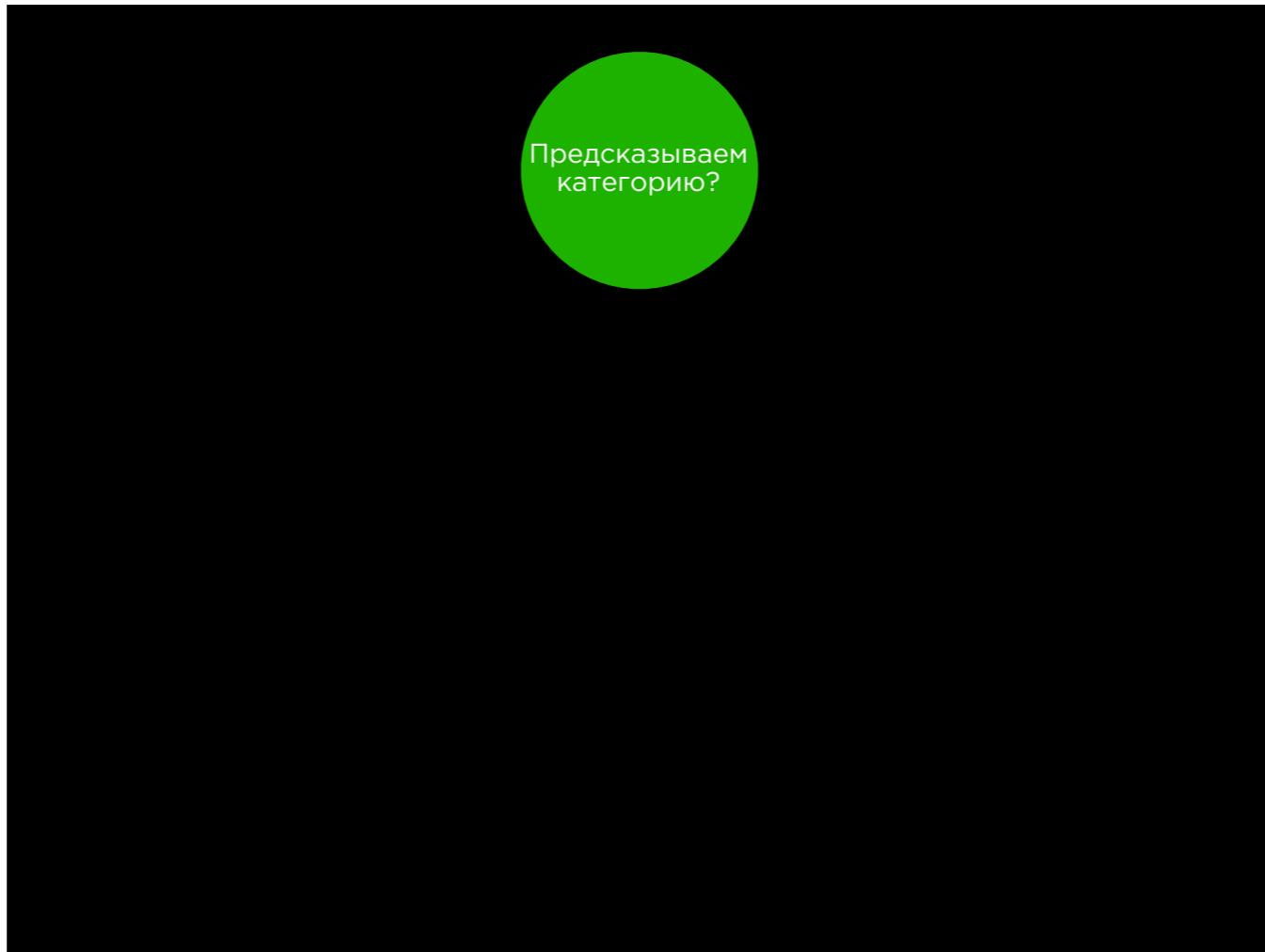
Но я решил выбрать эту, потому что она показывает выбор алгоритмов, без учета нейронных сетей.

По сути, к нейронным сетям стоит переходить тогда, когда есть ощущение комфорта в большинстве этих методов. В определенной степени потому что многие задачи не нуждаются в них. Картинки и текст — это задачи, которые могут и решаться с помощью нейронок, но частно нам попадаются табличные данные с разными категориальными и непрерывными переменными.

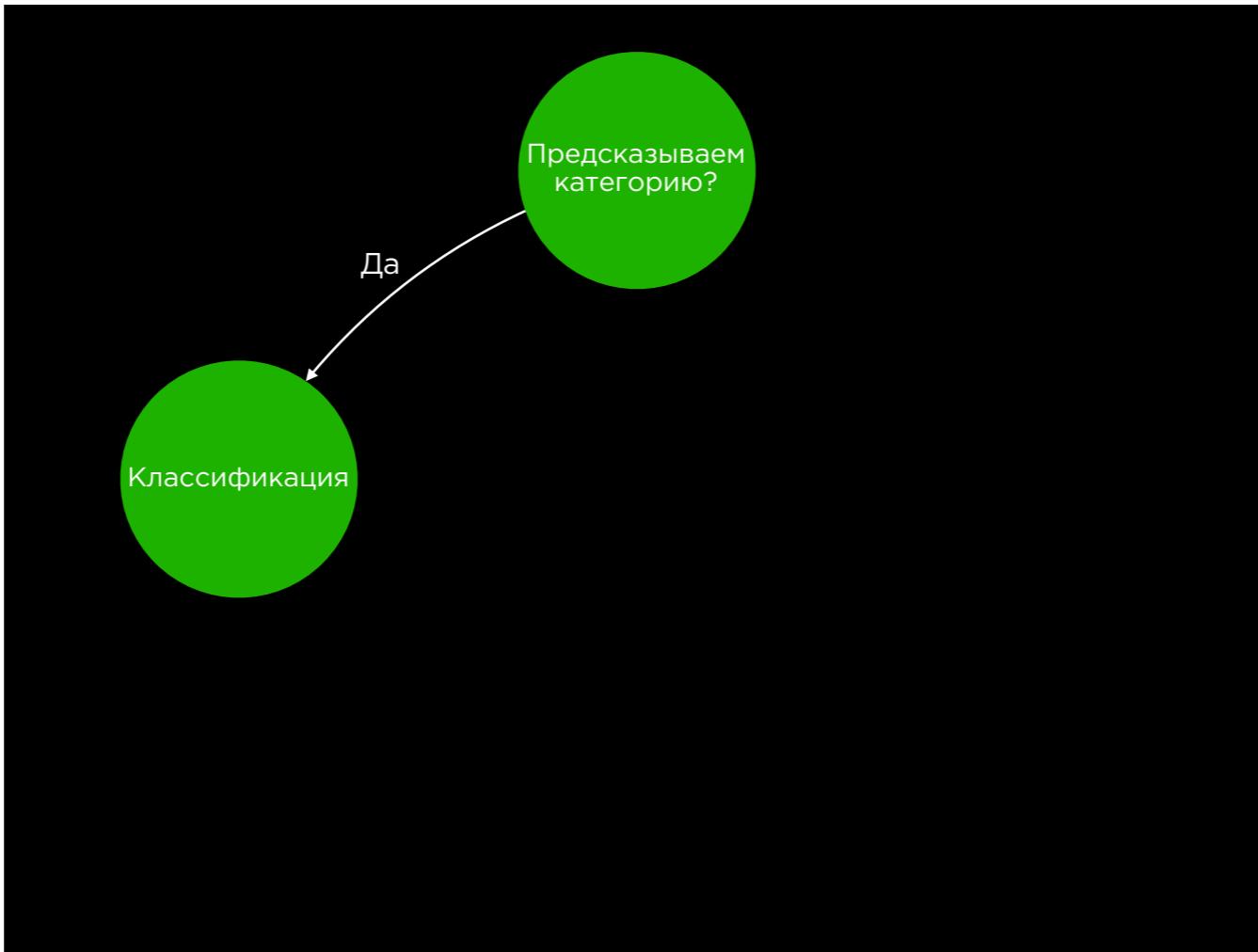
Помимо этого, как правило, мы пробуем разные *подходы (алгоритмы)* к решению задачи. И проще всего начинать всегда с sklearn-а. Так что вот.



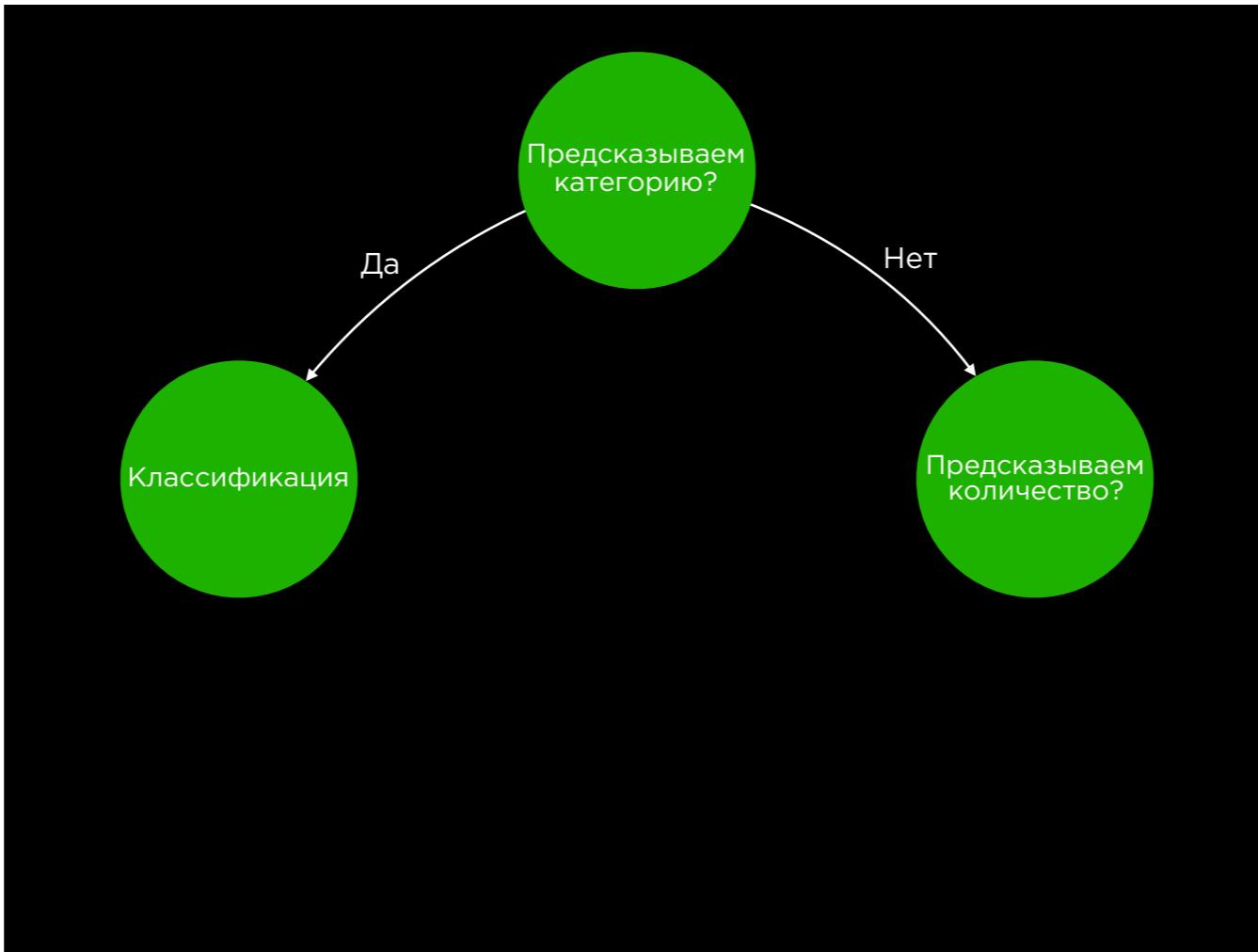
Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:



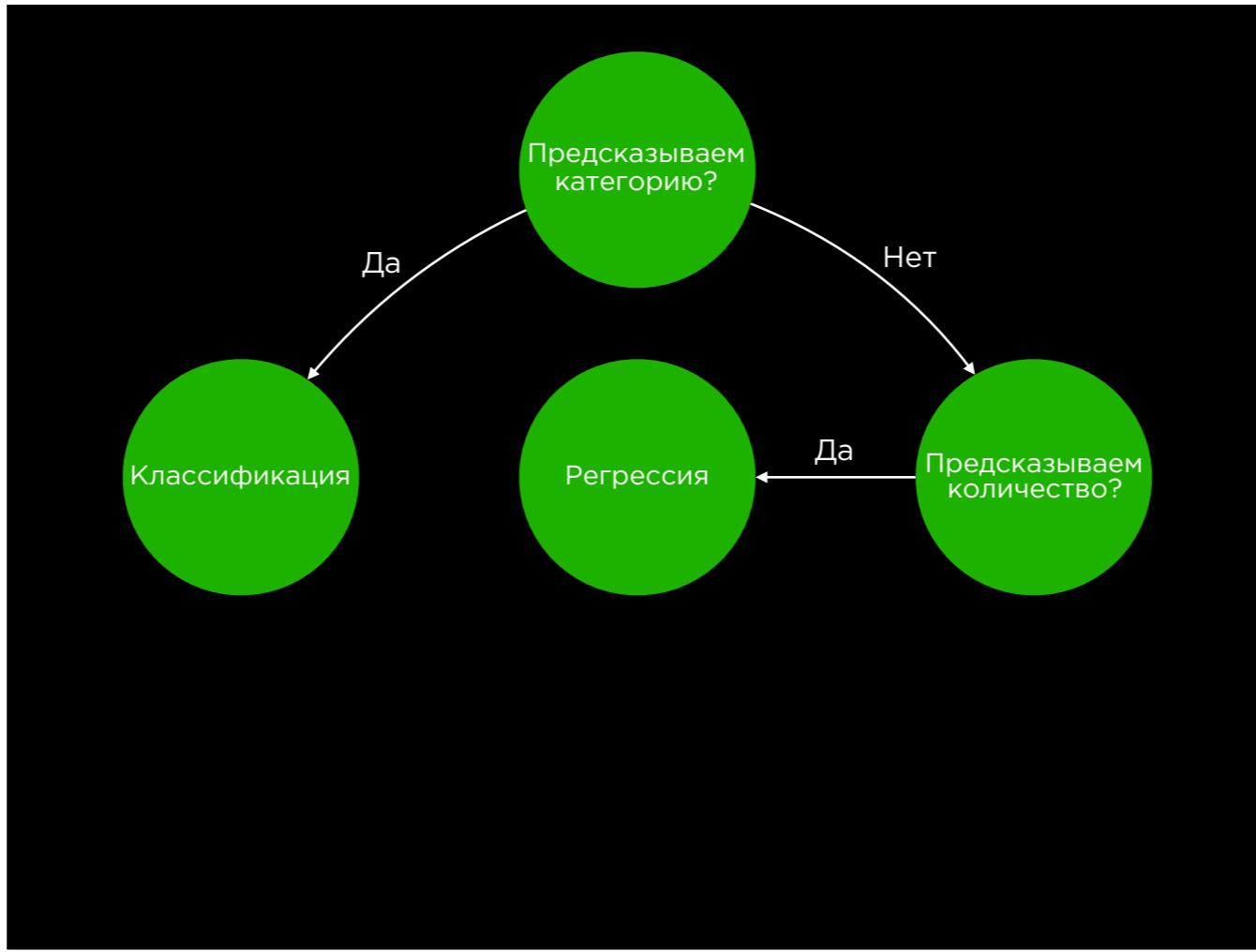
Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:



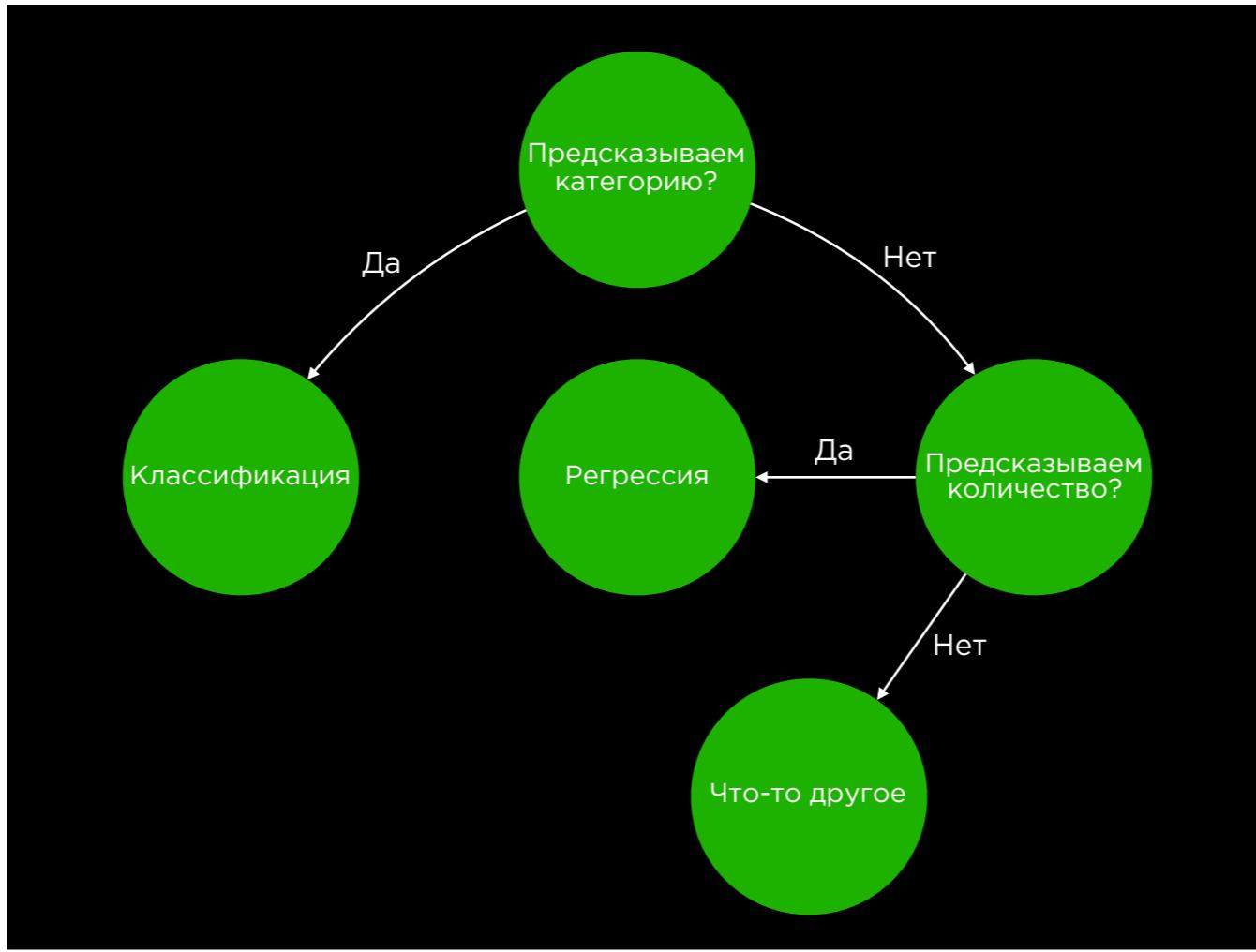
Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:



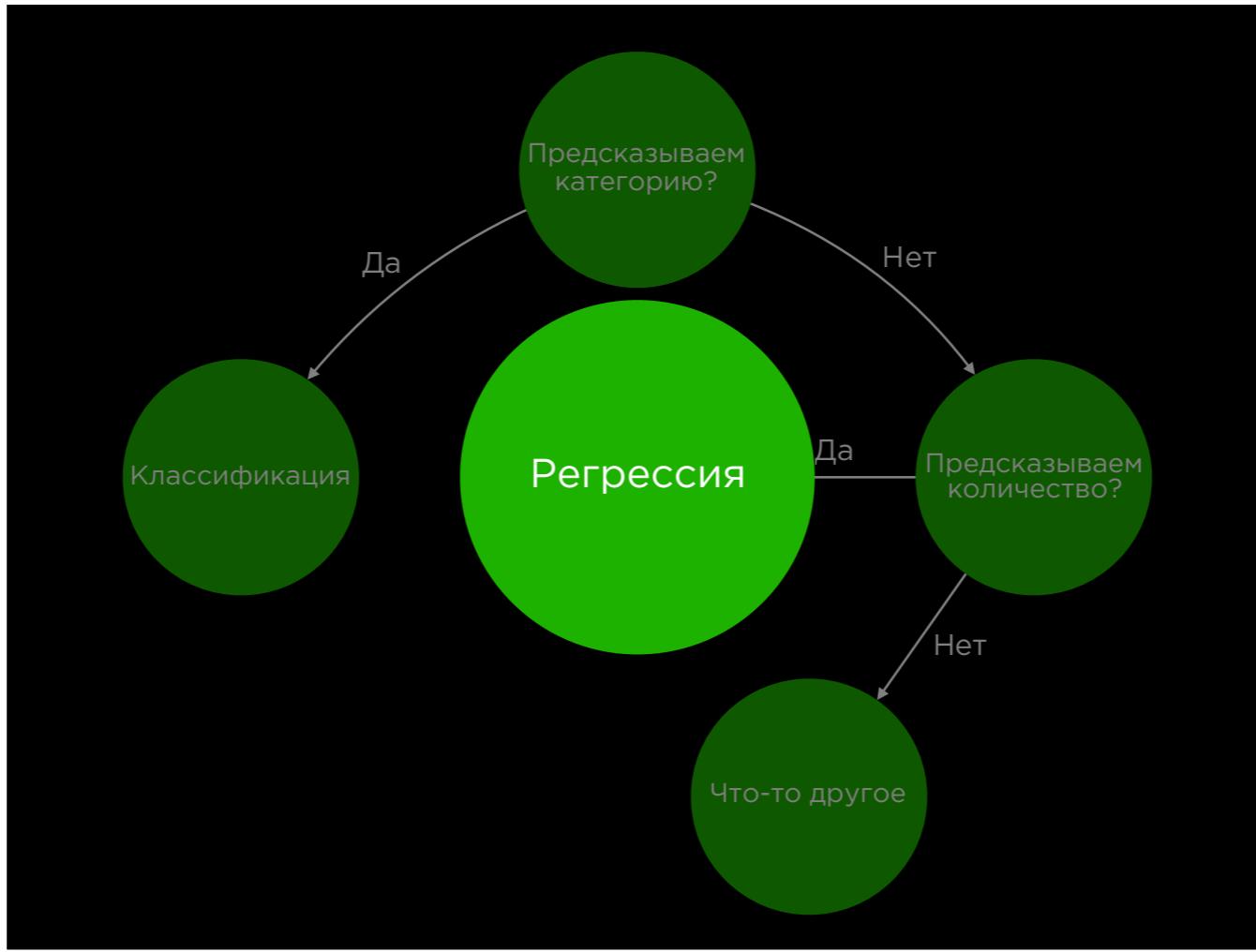
Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:



Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:



Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:



Но, если резюмировать ту картинку, то нас интересовать будут следующие вопросы:

Регрессия

Постановка задачи

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

Задача регрессии ставится примерно так

У нас есть некоторые значения X , от которых, мы думаем, зависит целевая переменная Y

И мы хотим, чтобы была или появилась какая-то такая функция f , которая по X -ам, говорила бы нам значение Y .

А еще есть нормальный шум.

Может быть вопрос, зачем вообще этот нормальный шум?

Это слагаемое закладывается статистиками, чтобы предусмотреть наблюдения, которых у нас нет.

Постановка задачи

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

Задача регрессии ставится примерно так

У нас есть некоторые значения X , от которых, мы думаем, зависит целевая переменная Y

И мы хотим, чтобы была или появилась какая-то такая функция f , которая по X -ам, говорила бы нам значение Y .

А еще есть нормальный шум.

Может быть вопрос, зачем вообще этот нормальный шум?

Это слагаемое закладывается статистиками, чтобы предусмотреть наблюдения, которых у нас нет.

Постановка задачи

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

независимые
переменные

Задача регрессии ставится примерно так

У нас есть некоторые значения X , от которых, мы думаем, зависит целевая переменная Y

И мы хотим, чтобы была или появилась какая-то такая функция f , которая по X -ам, говорила бы нам значение Y .

А еще есть нормальный шум.

Может быть вопрос, зачем вообще этот нормальный шум?

Это слагаемое закладывается статистиками, чтобы предусмотреть наблюдения, которых у нас нет.

Постановка задачи



Задача регрессии ставится примерно так

У нас есть некоторые значения X , от которых, мы думаем, зависит целевая переменная Y

И мы хотим, чтобы была или появилась какая-то такая функция f , которая по X -ам, говорила бы нам значение Y .

А еще есть нормальный шум.

Может быть вопрос, зачем вообще этот нормальный шум?

Это слагаемое закладывается статистиками, чтобы предусмотреть наблюдения, которых у нас нет.

Постановка задачи



Задача регрессии ставится примерно так

У нас есть некоторые значения X , от которых, мы думаем, зависит целевая переменная Y

И мы хотим, чтобы была или появилась какая-то такая функция f , которая по X -ам, говорила бы нам значение Y .

А еще есть нормальный шум.

Может быть вопрос, зачем вообще этот нормальный шум?

Это слагаемое закладывается статистиками, чтобы предусмотреть наблюдения, которых у нас нет.

Постановка задачи



Задача регрессии ставится примерно так

У нас есть некоторые значения X , от которых, мы думаем, зависит целевая переменная Y

И мы хотим, чтобы была или появилась какая-то такая функция f , которая по X -ам, говорила бы нам значение Y .

А еще есть нормальный шум.

Может быть вопрос, зачем вообще этот нормальный шум?

Это слагаемое закладывается статистиками, чтобы предусмотреть наблюдения, которых у нас нет.

Самое простое F?

В самом простом случае, чем будет F?

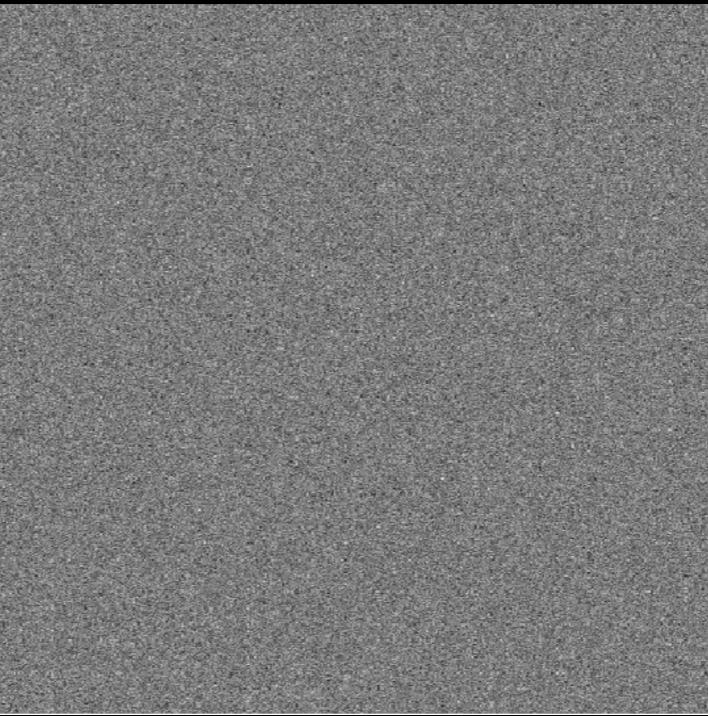
Константа!

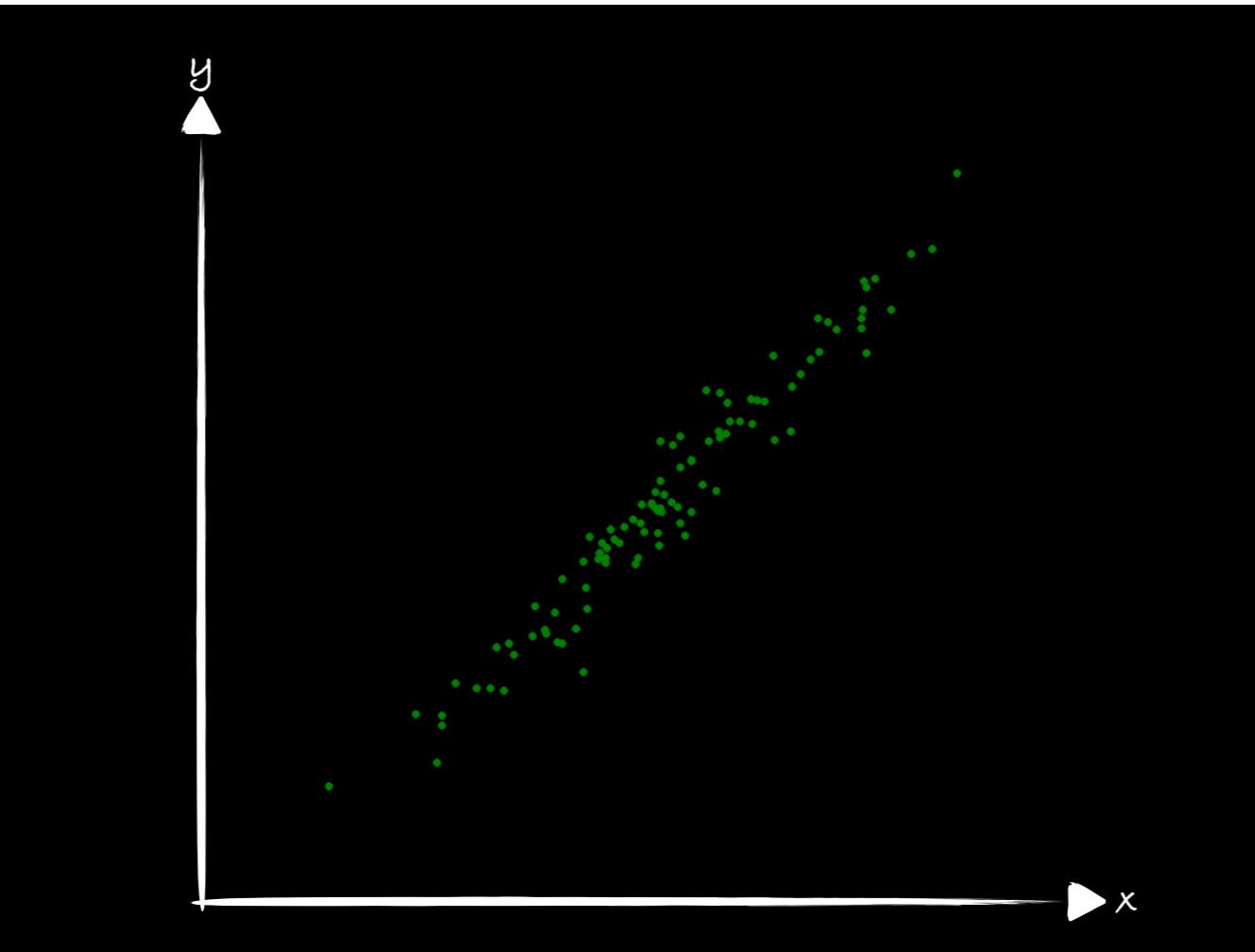
$$f(X) = 5$$

И в каких-то случаях, мы угадаем, в каких-то нет

Сложнее, но так же бесполезно?

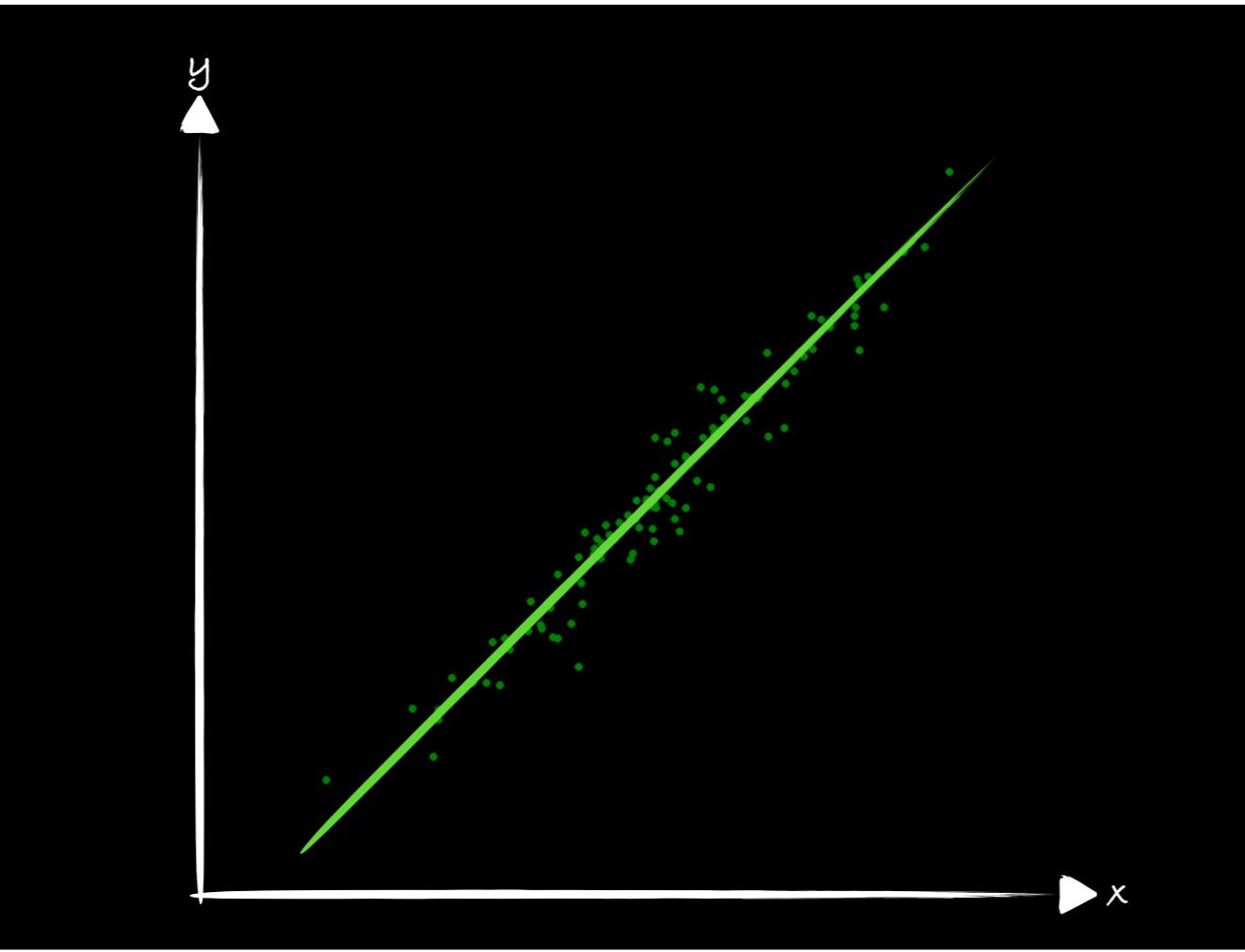
$$f(X) =$$





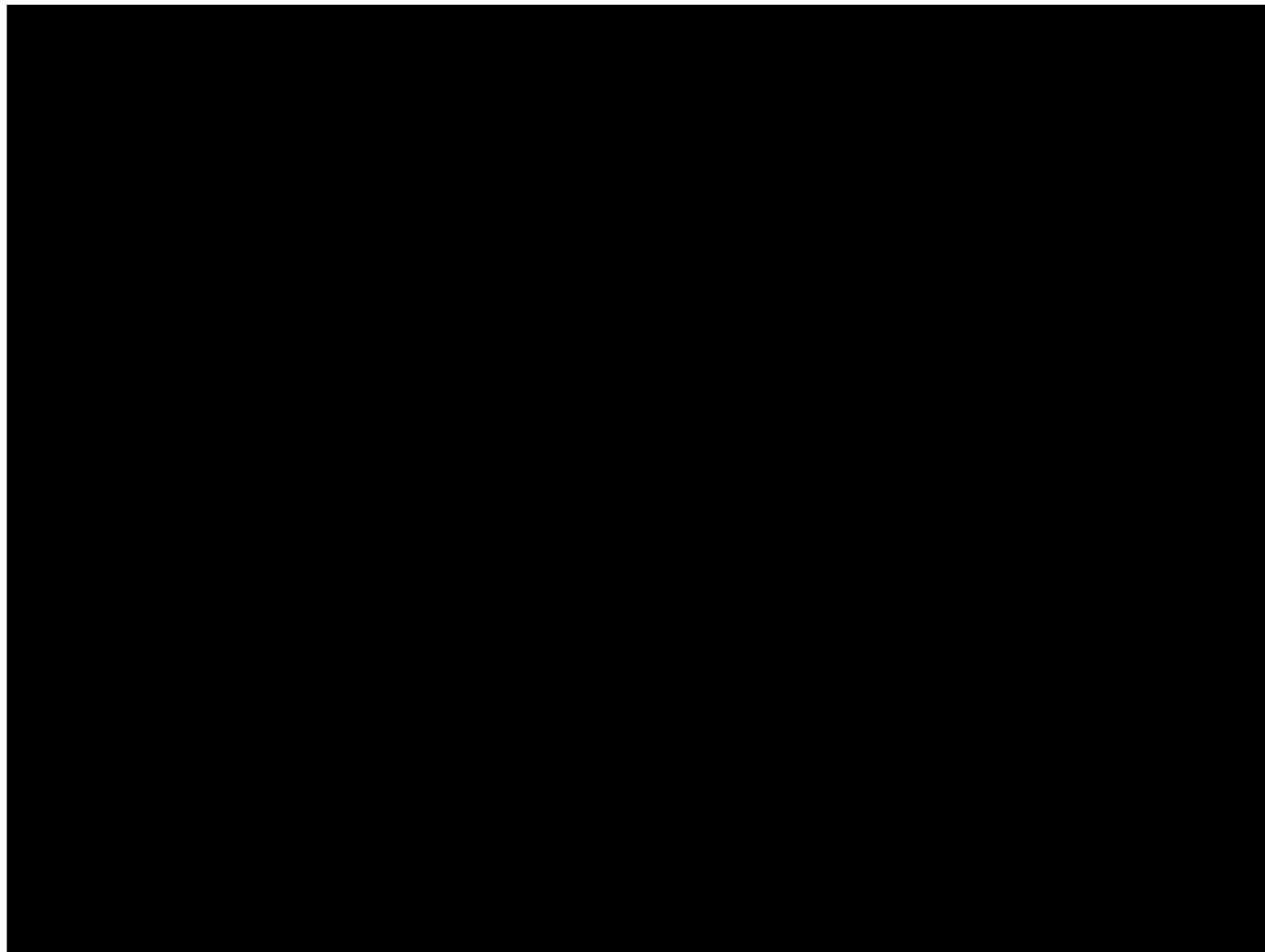
Но окей, а что мы бы, на самом деле хотели бы сделать, если бы увидели такую картину?

Правильно! Мы бы хотели просто нарисовать прямую через эти точки.



Но окей, а что мы бы, на самом деле хотели бы сделать, если бы увидели такую картину?

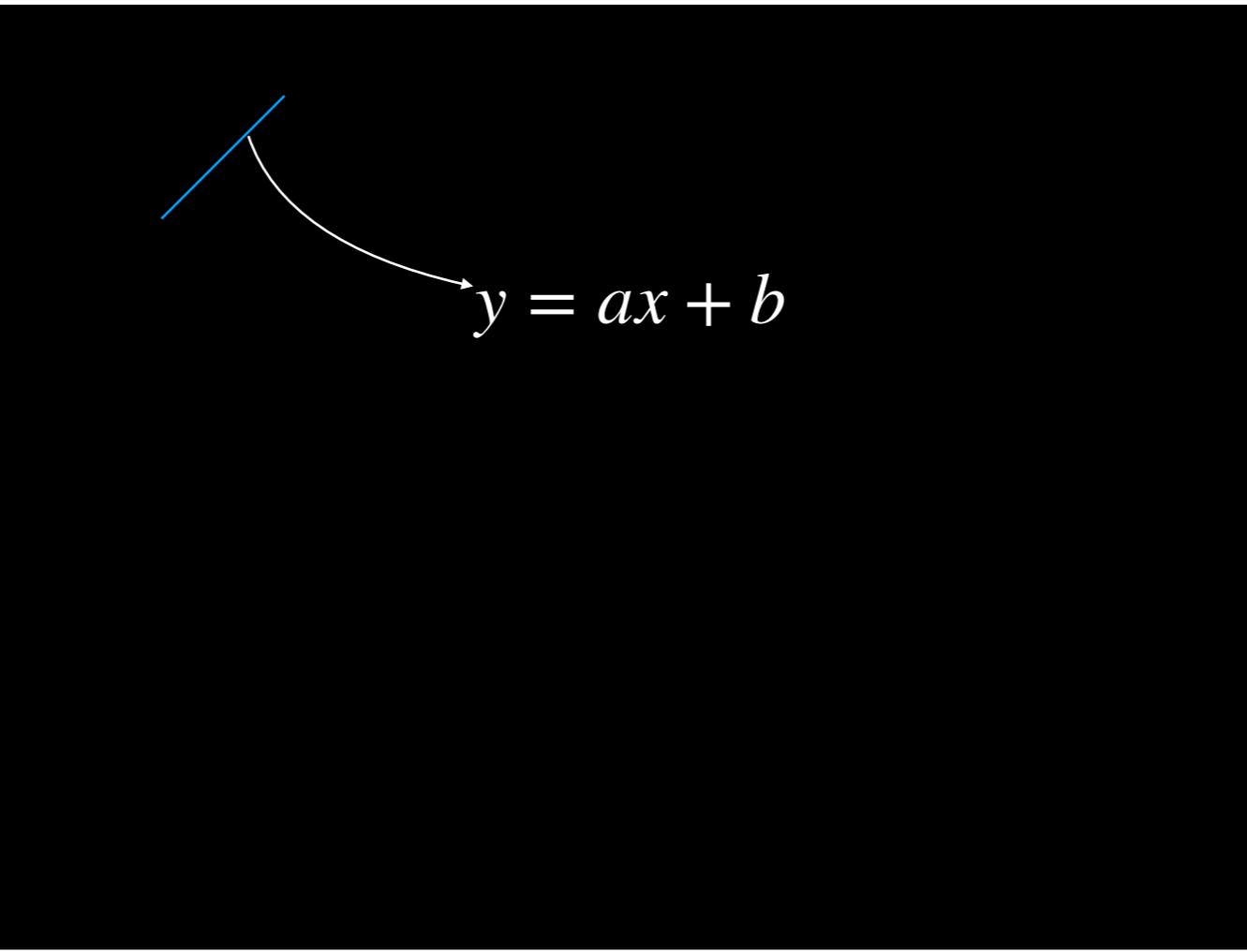
Правильно! Мы бы хотели просто нарисовать прямую через эти точки.



В таком случае, получается, если нам повезло, и надо просто смоделить что-то около прямой, то нам хватит линии

А кто вспомнит уравнение прямой?

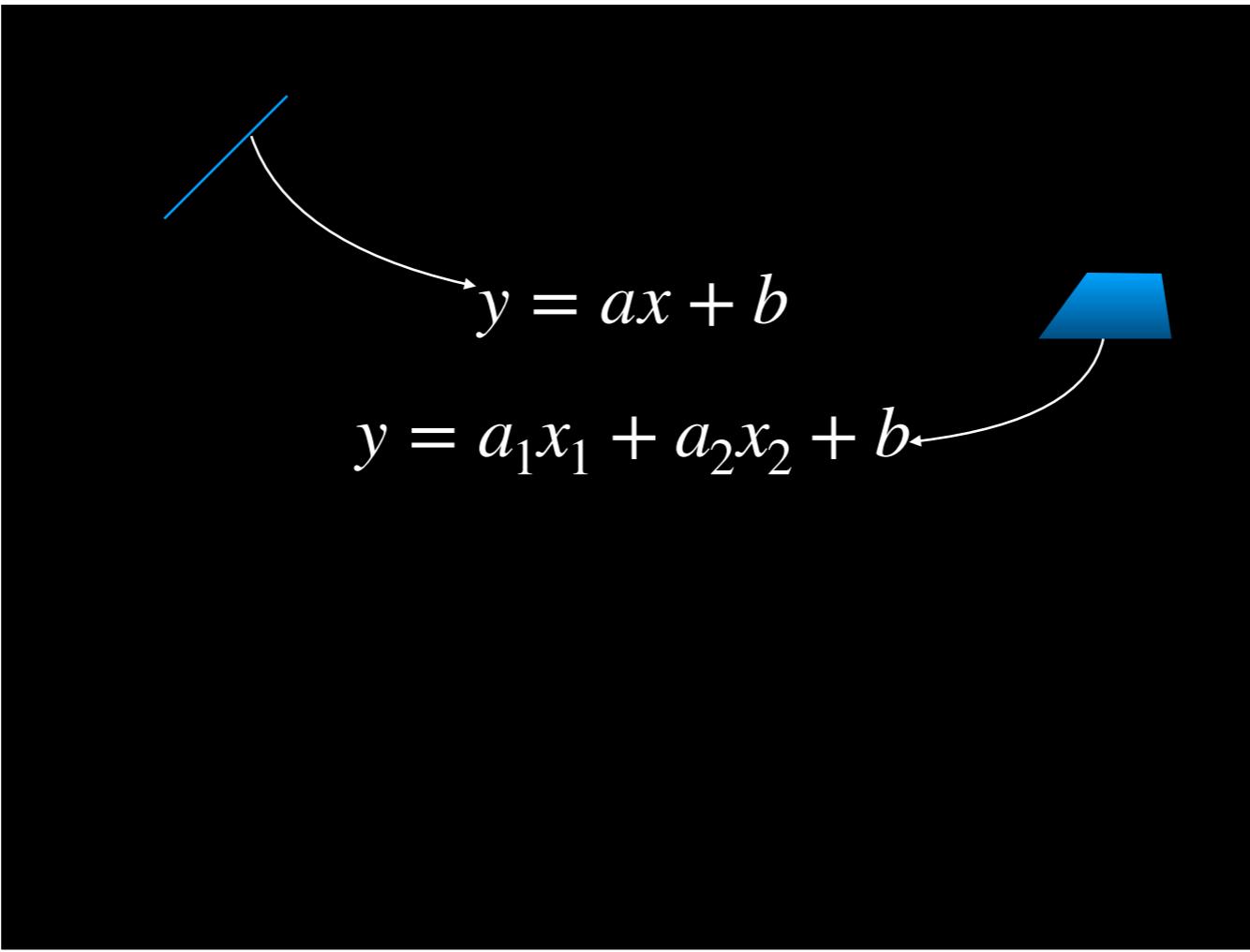
А у равнение прямой, я напомню вам



В таком случае, получается, если нам повезло, и надо просто смоделиить что-то около прямой, то нам хватит линии

А кто вспомнит уравнение прямой?

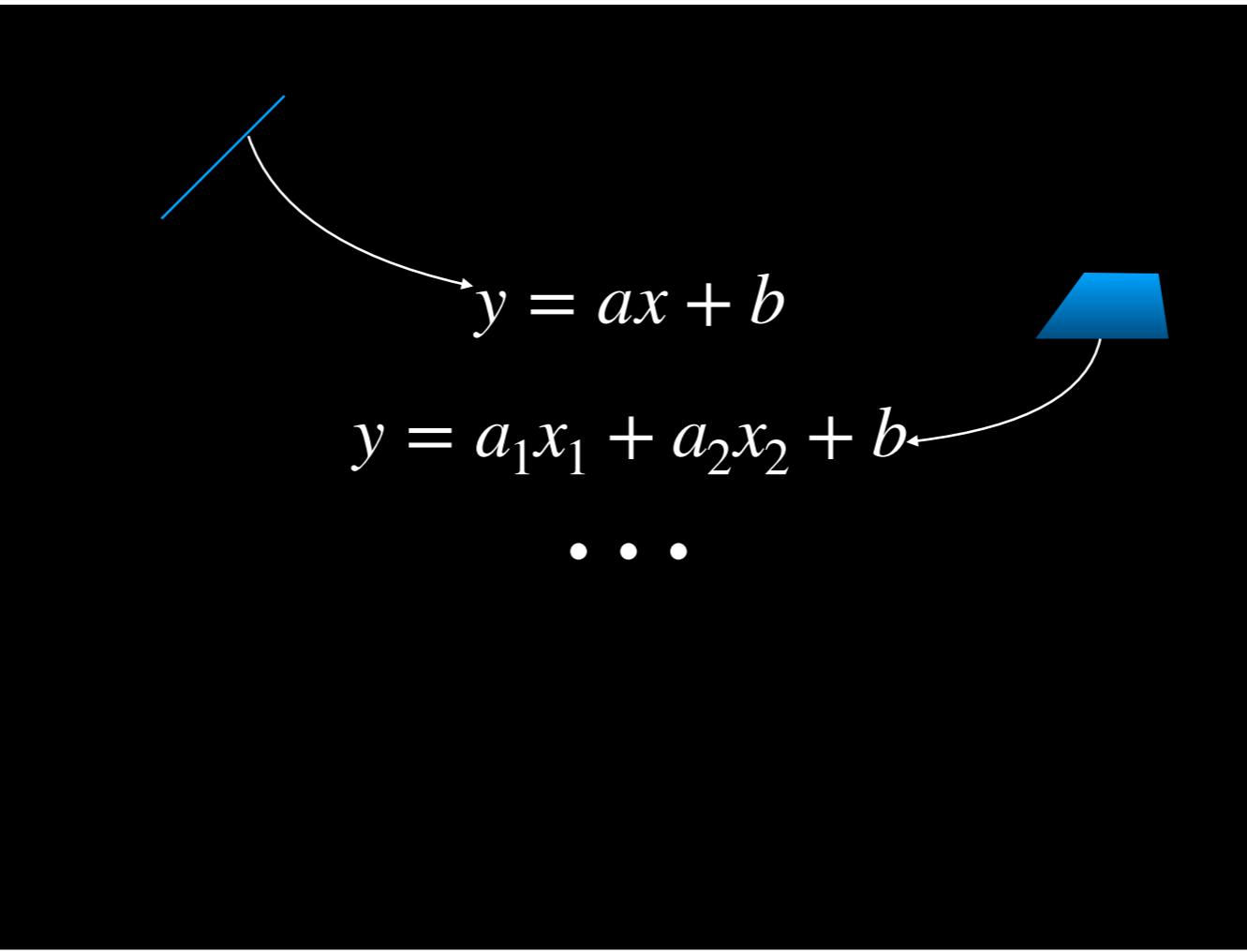
А у равнение прямой, я напомню вам



В таком случае, получается, если нам повезло, и надо просто смоделировать что-то около прямой, то нам хватит линии

А кто вспомнит уравнение прямой?

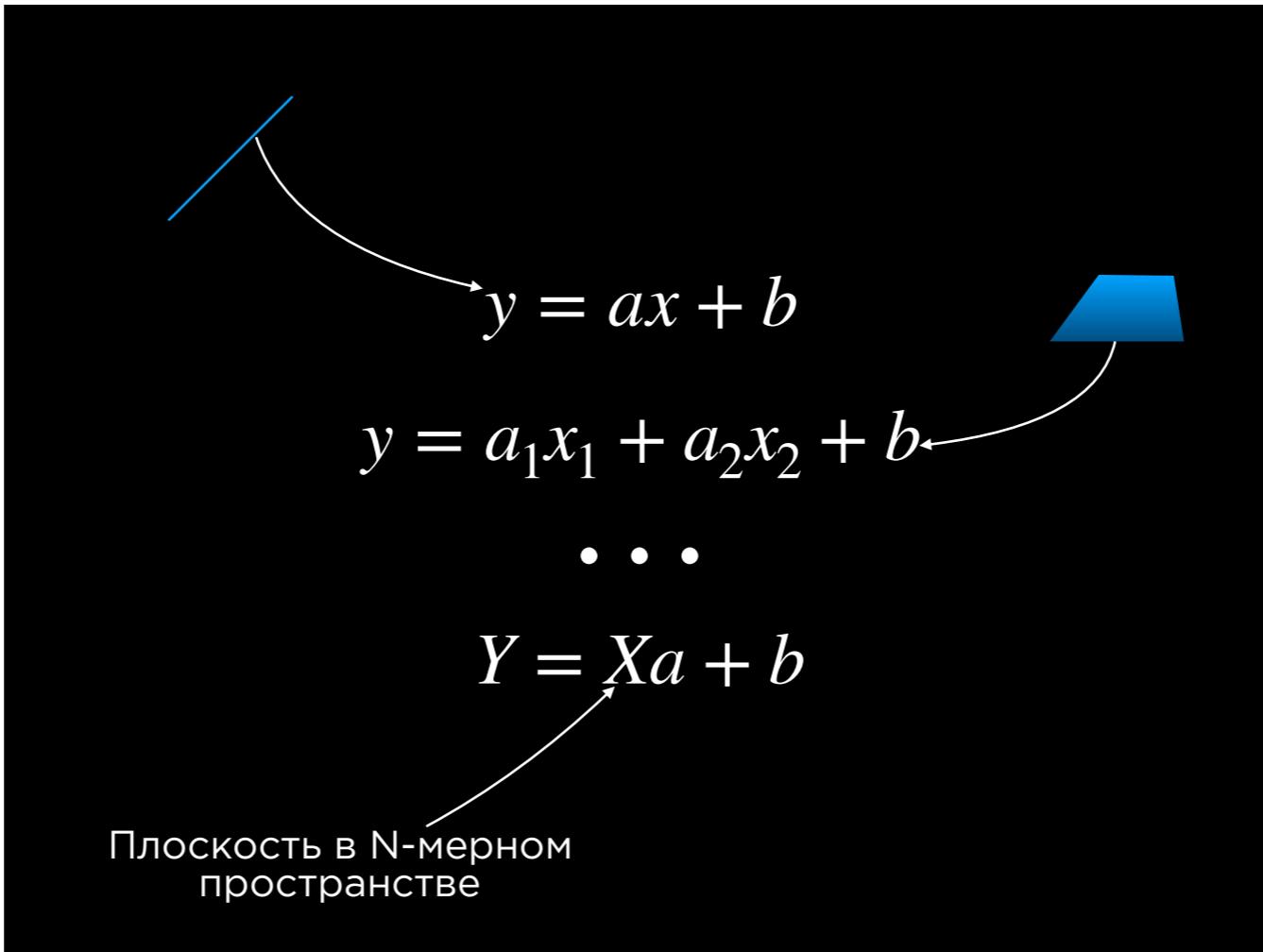
А у равнение прямой, я напомню вам



В таком случае, получается, если нам повезло, и надо просто смоделировать что-то около прямой, то нам хватит линии

А кто вспомнит уравнение прямой?

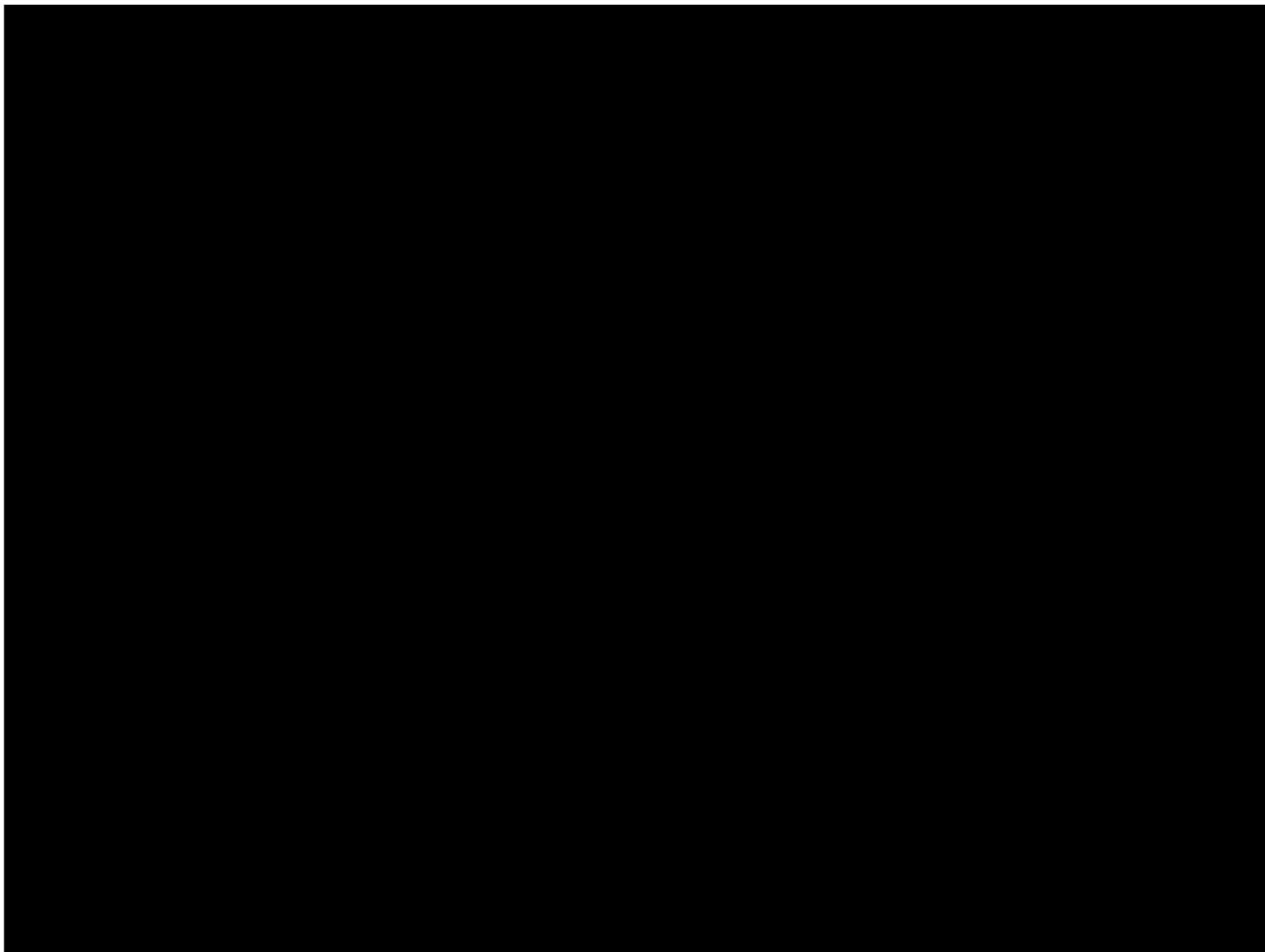
А у равнение прямой, я напомню вам



В таком случае, получается, если нам повезло, и надо просто смоделировать что-то около прямой, то нам хватит линии

А кто вспомнит уравнение прямой?

А у равнение прямой, я напомню вам



Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

Ну вы взяли эти данные и построили график

Магистратура

Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

Ну вы взяли эти данные и построили график

Магистратура



Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

Ну вы взяли эти данные и построили график

Магистратура

Ожидаемые баллы



Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

Ну вы взяли эти данные и построили график

Магистратура

Ожидаемые баллы

Результаты прошлых лет



Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

Ну вы взяли эти данные и построили график

Магистратура

Ожидаемые баллы

Результаты прошлых лет

Количество мест



Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

Ну вы взяли эти данные и построили график

Магистратура

Ожидаемые баллы



Результаты прошлых лет

Количество мест



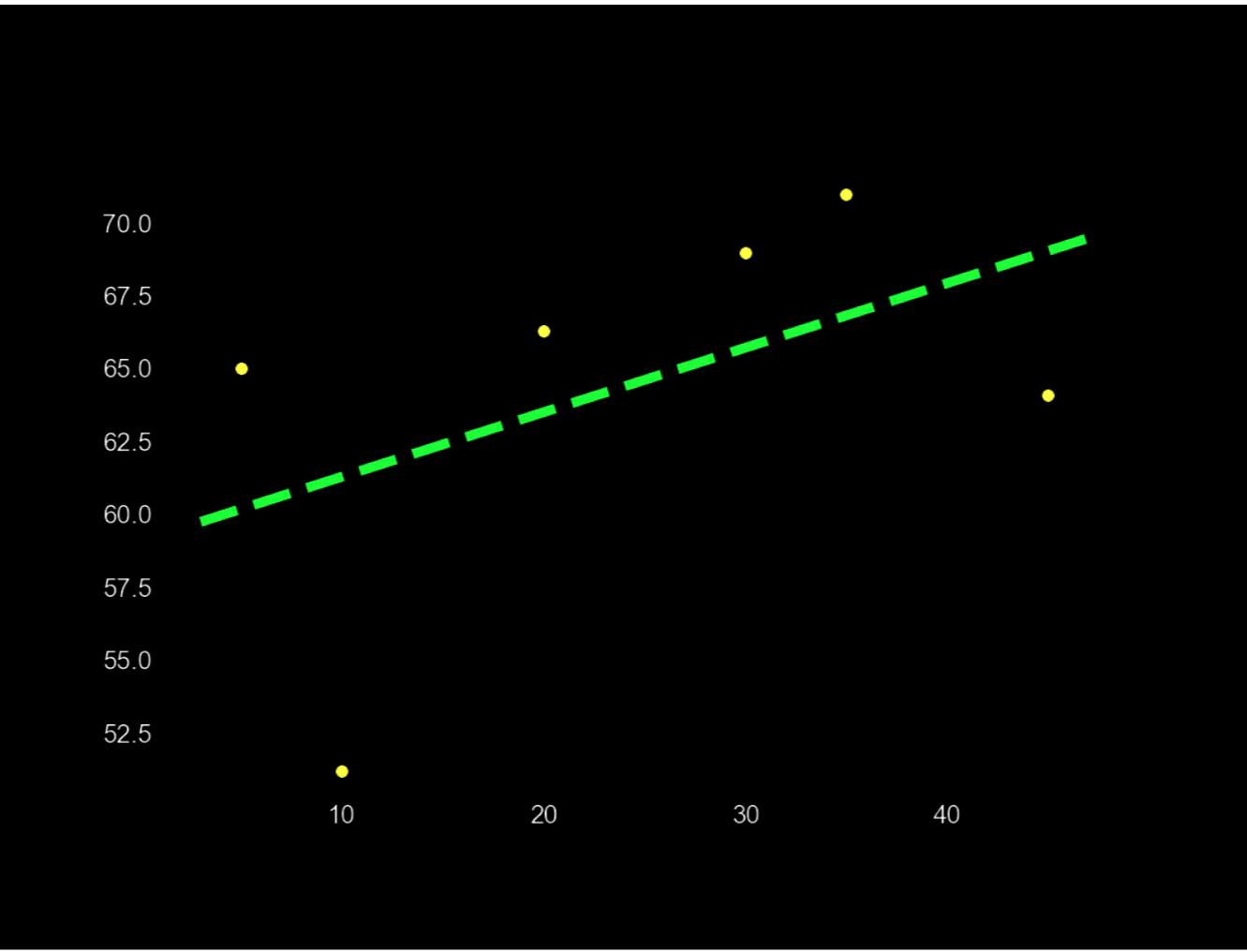
Чтобы разбавить эту нудятину, я приведу вам пример из реальной жизни.

Предположим, вы хотите поступать в магистратуру

И вы смотрели всякие документы, баллы прошлых лет и у вас имеется информация, о том, сколько вы можете ожидать баллов при поступлении и исторические данные о баллах. К сожалению, в вашем распоряжении есть только инфа о баллах прошлых лет и изменении количества мест.

И вам пришла в голову мысль, что может быть есть какая-то зависимость между баллами и количеством мест?

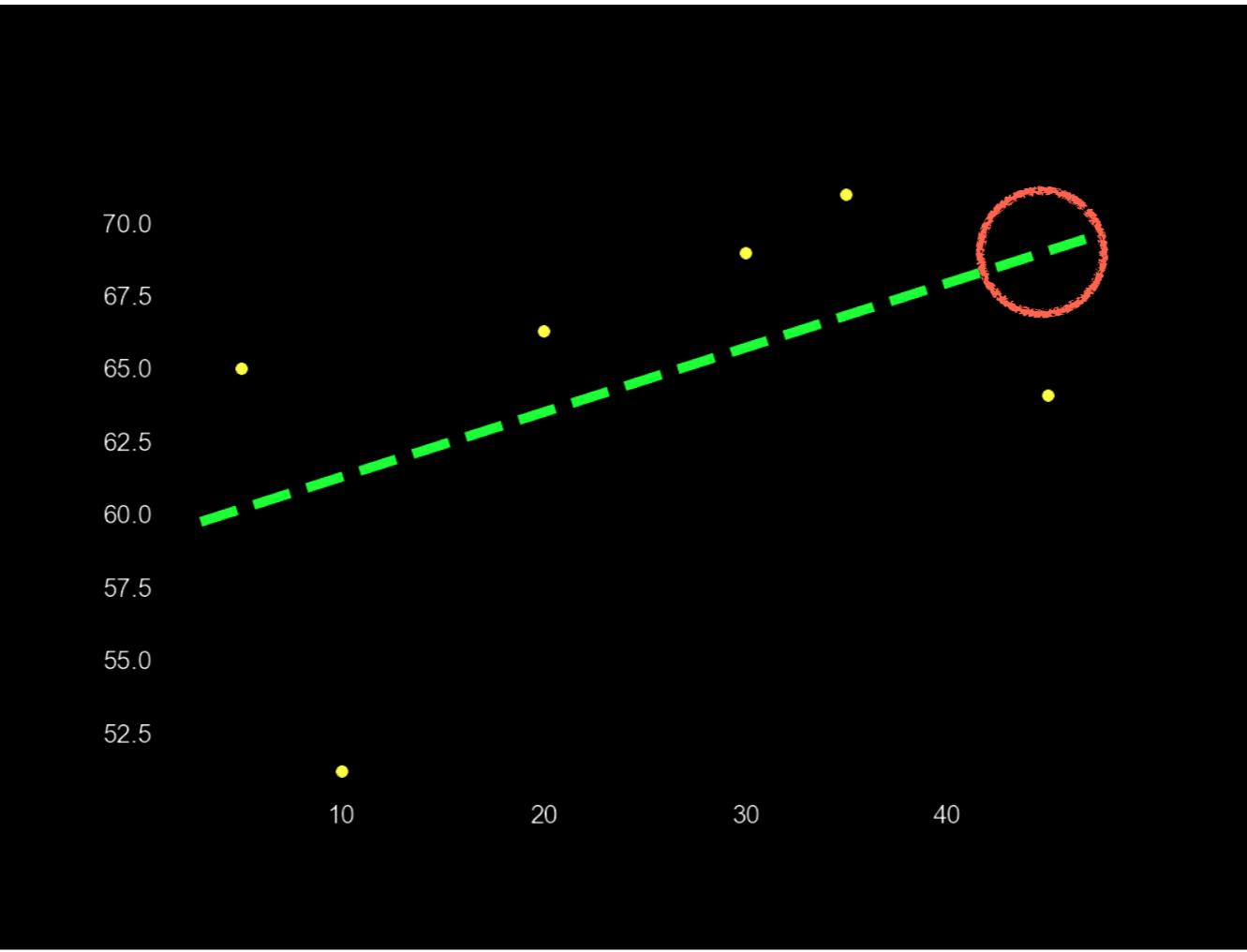
Ну вы взяли эти данные и построили график



И построенная модель будет выглядеть как-то так

А теперь, мы знаем, что в этом году мест на матобессе будет 45

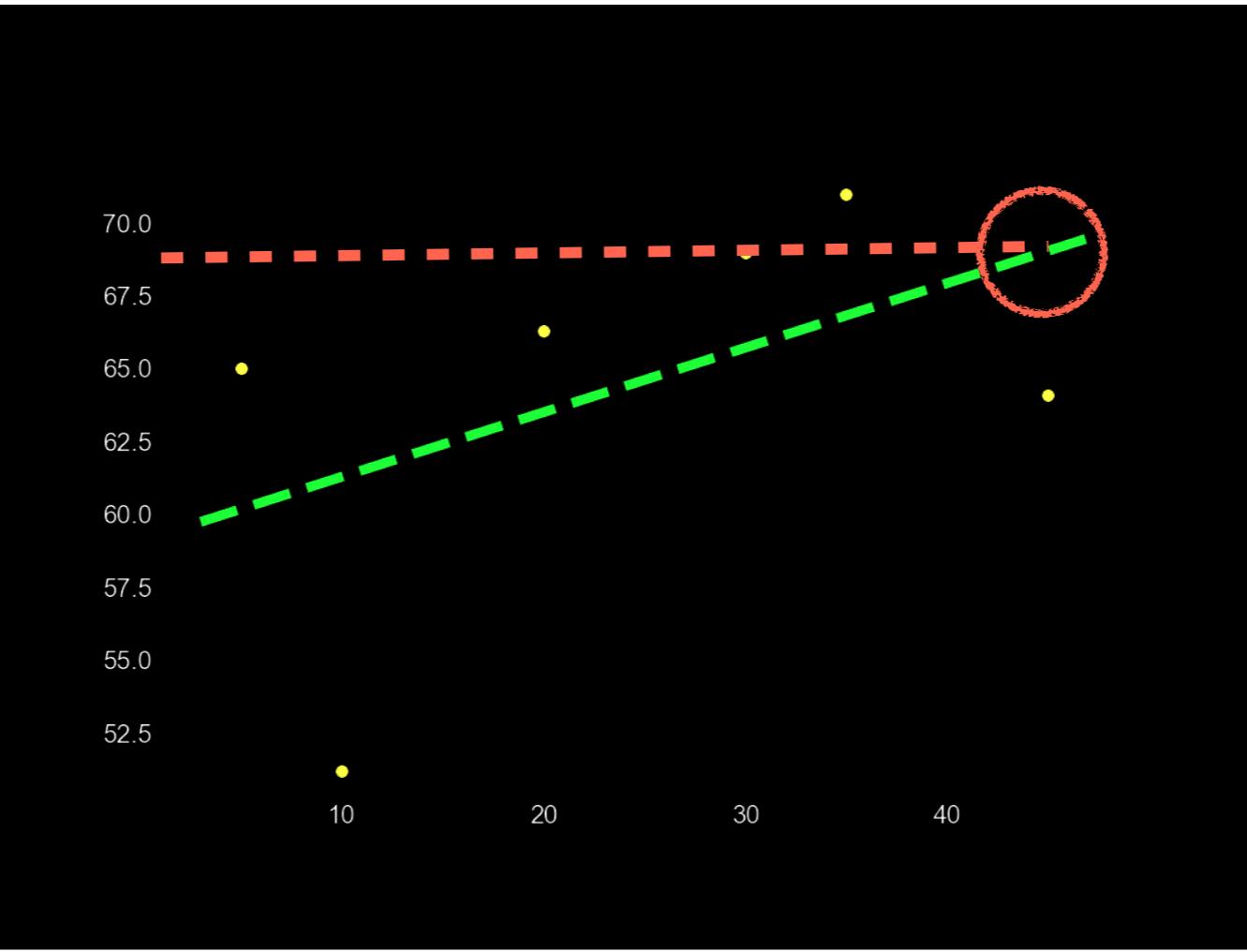
И мы можем задать вопрос модели: какой средний балл ожидается в этом году?



И построенная модель будет выглядеть как-то так

А теперь, мы знаем, что в этом году мест на матобессе будет 45

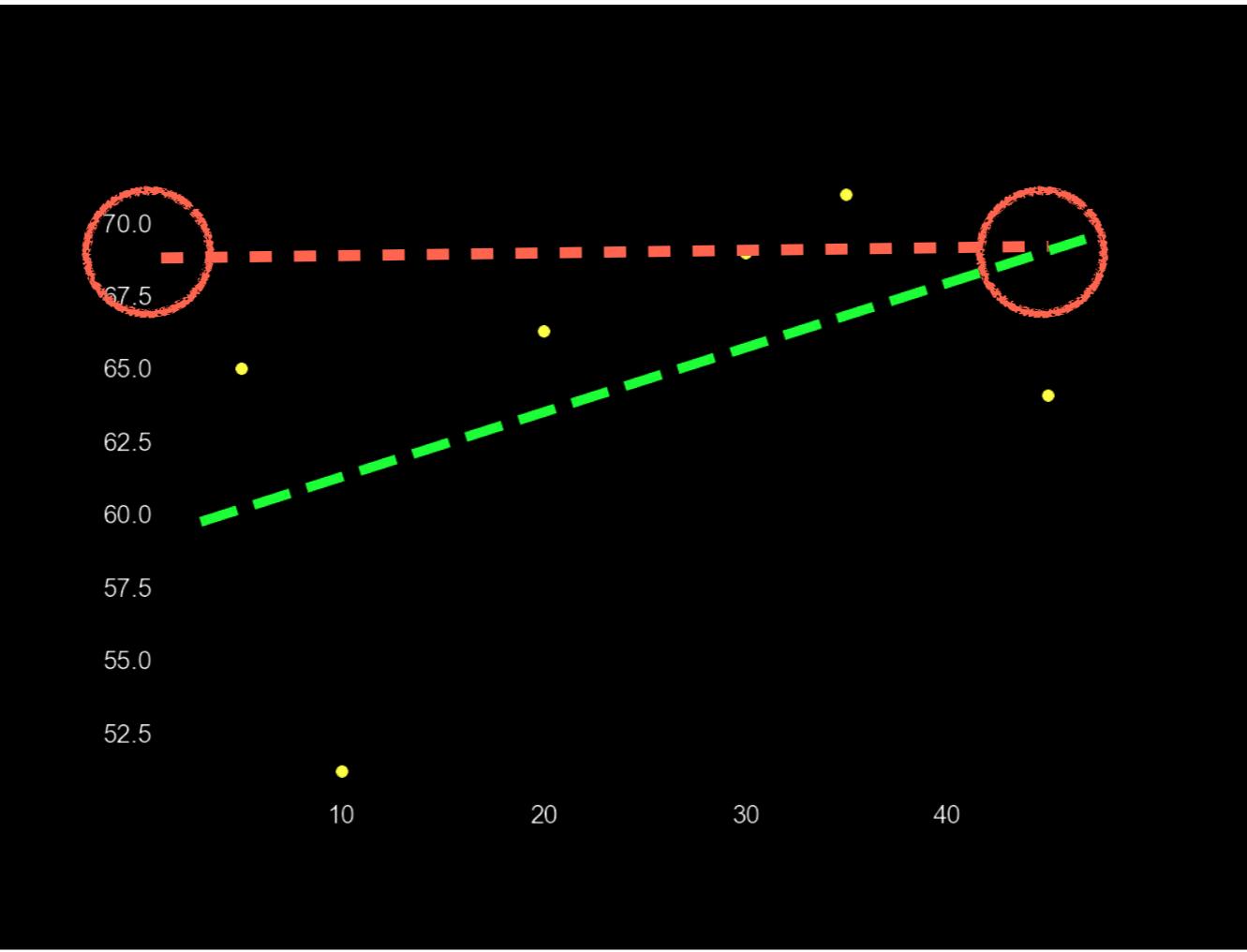
И мы можем задать вопрос модели: какой средний балл ожидается в этом году?



И построенная модель будет выглядеть как-то так

А теперь, мы знаем, что в этом году мест на матобессе будет 45

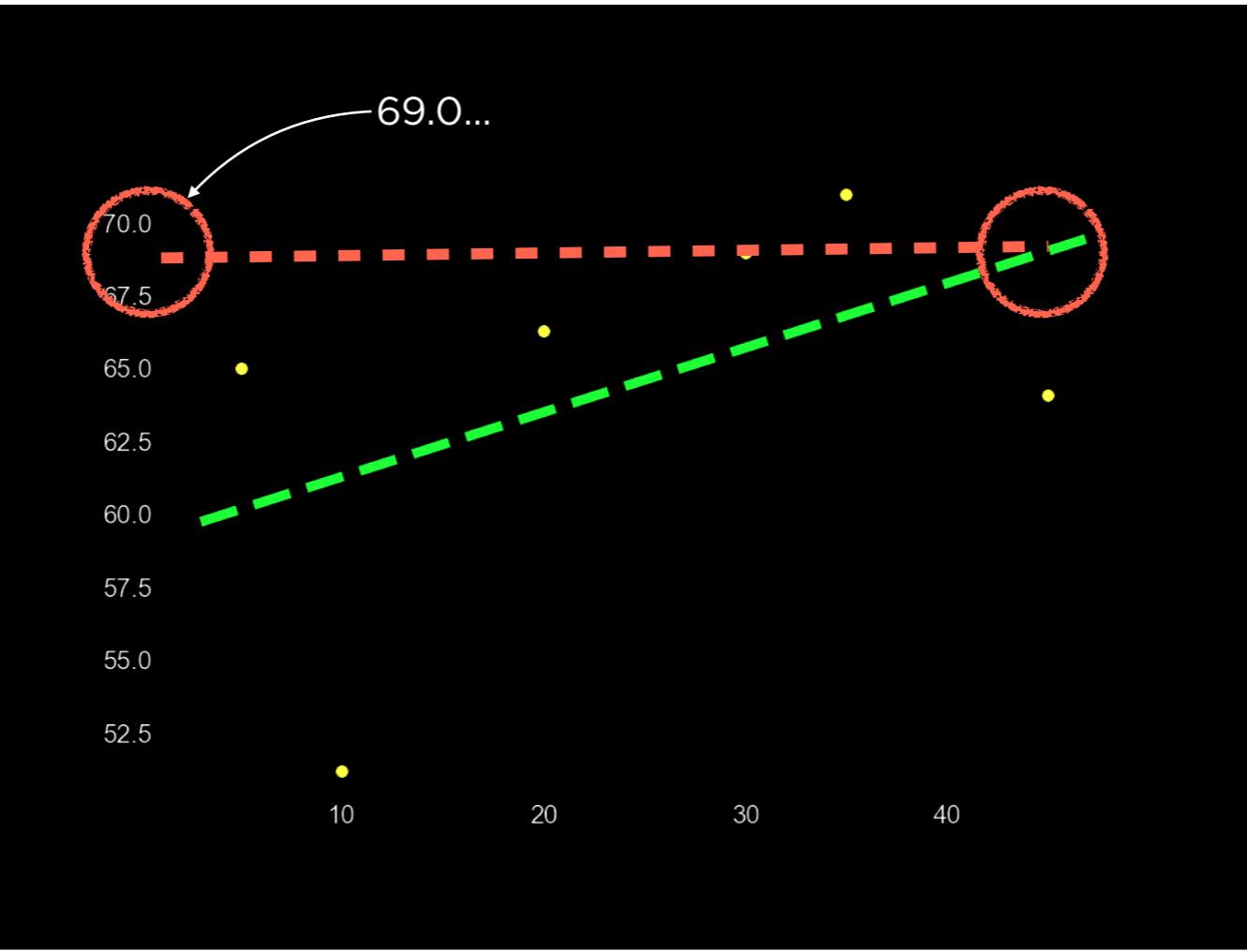
И мы можем задать вопрос модели: какой средний балл ожидается в этом году?



И построенная модель будет выглядеть как-то так

А теперь, мы знаем, что в этом году мест на матобессе будет 45

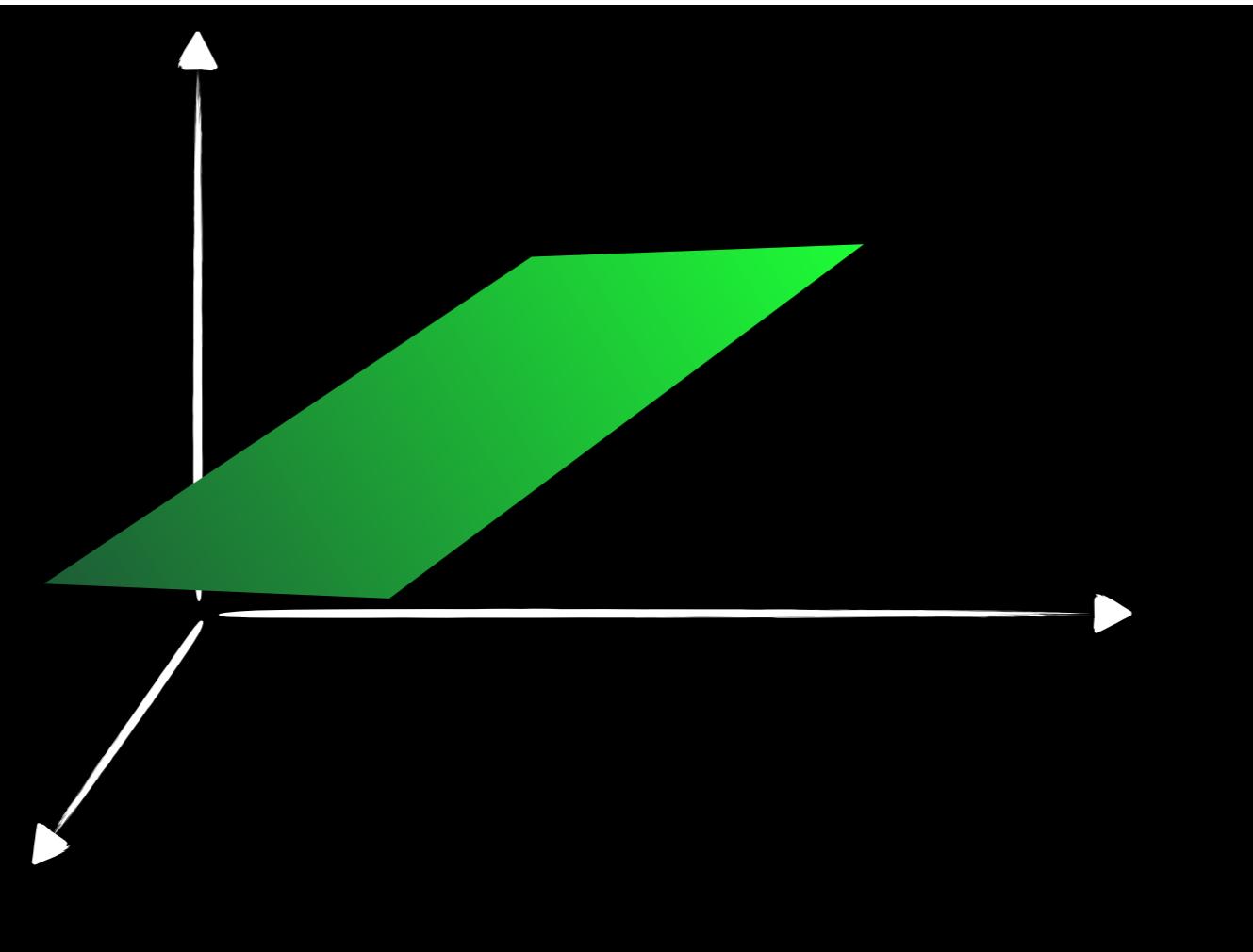
И мы можем задать вопрос модели: какой средний балл ожидается в этом году?



И построенная модель будет выглядеть как-то так

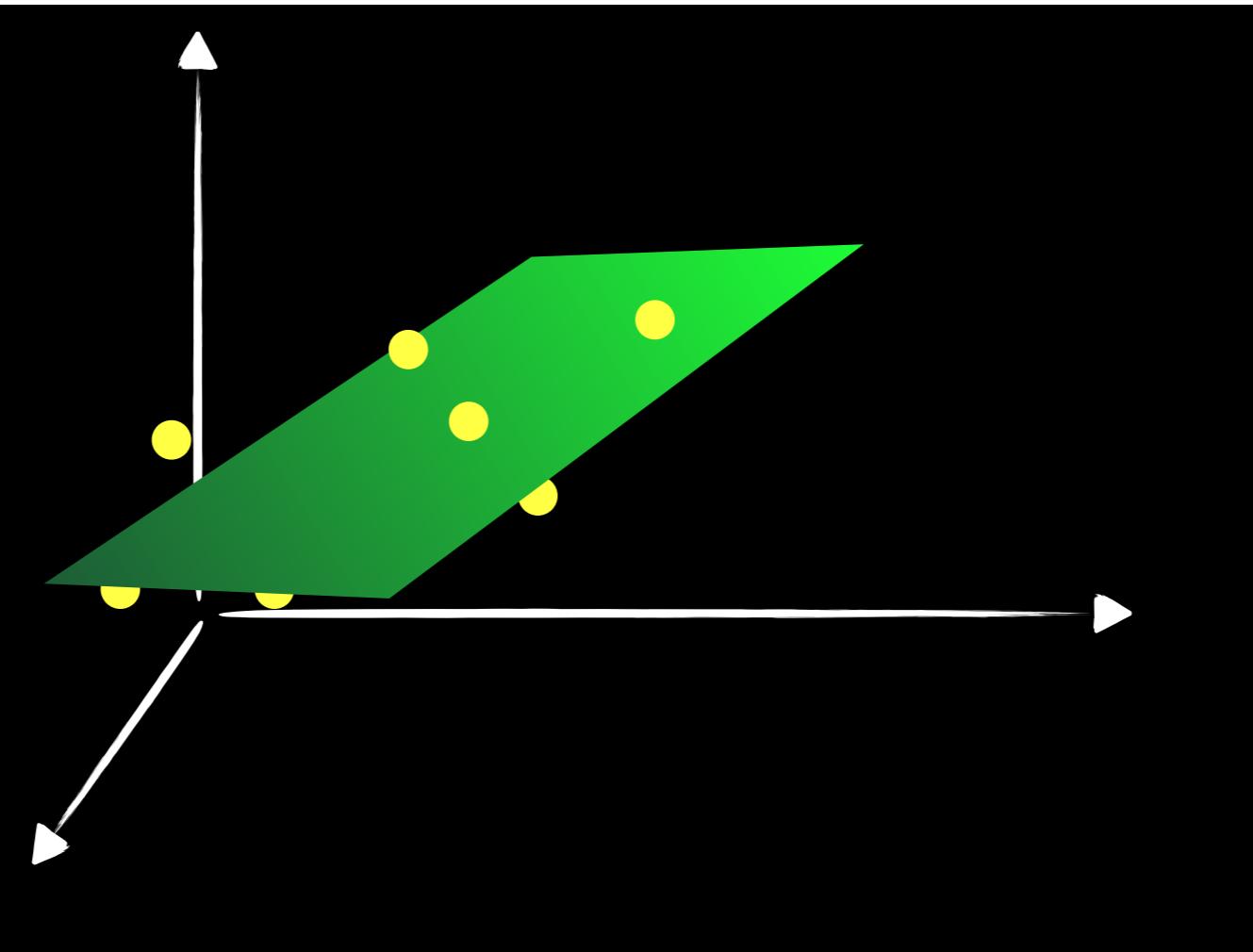
А теперь, мы знаем, что в этом году мест на матобессе будет 45

И мы можем задать вопрос модели: какой средний балл ожидается в этом году?



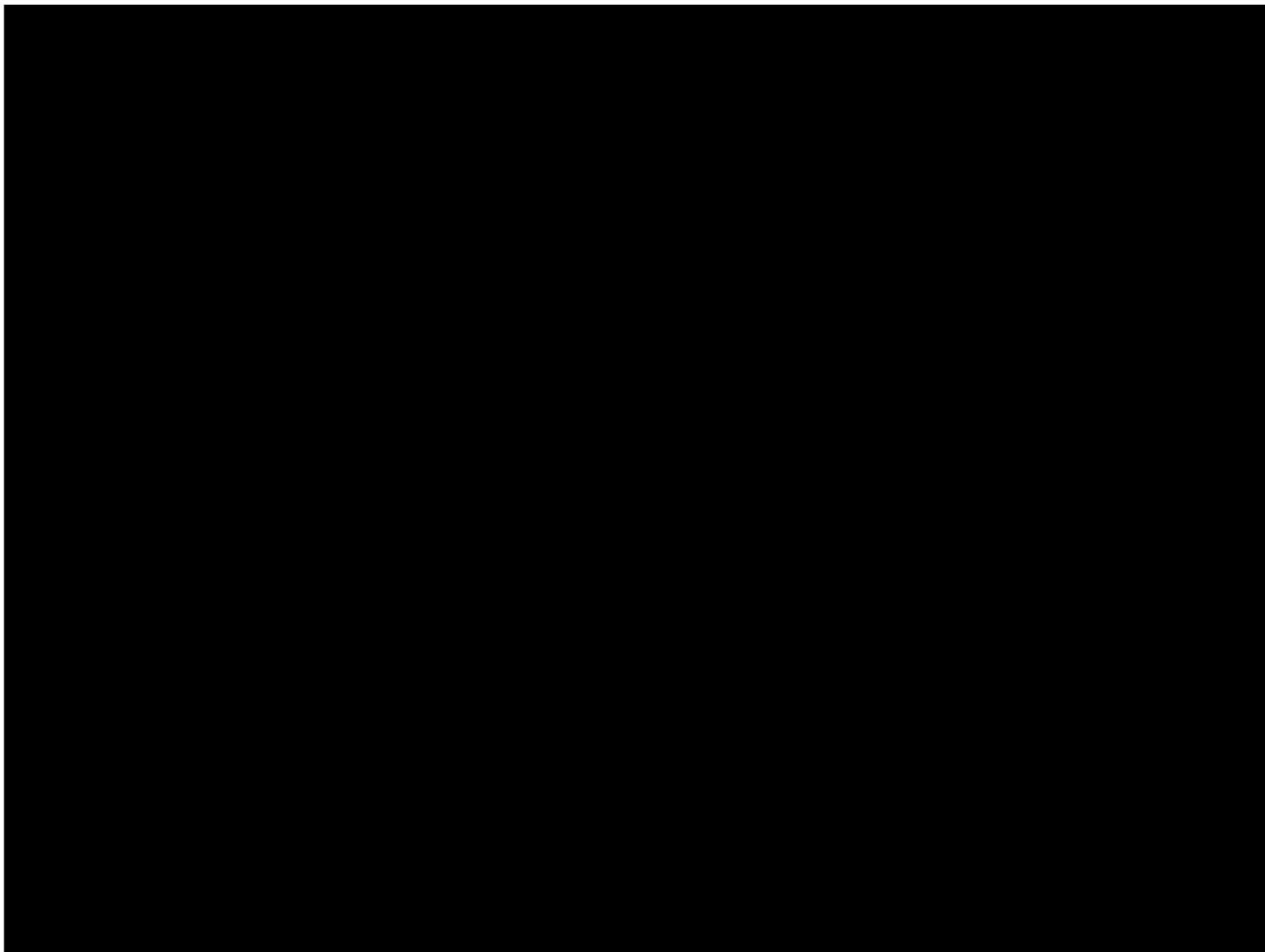
В многомерном случае у нас была бы такая плоскость, которую мы бы построили используя эти точки.

Ну и так далее по размерностям.



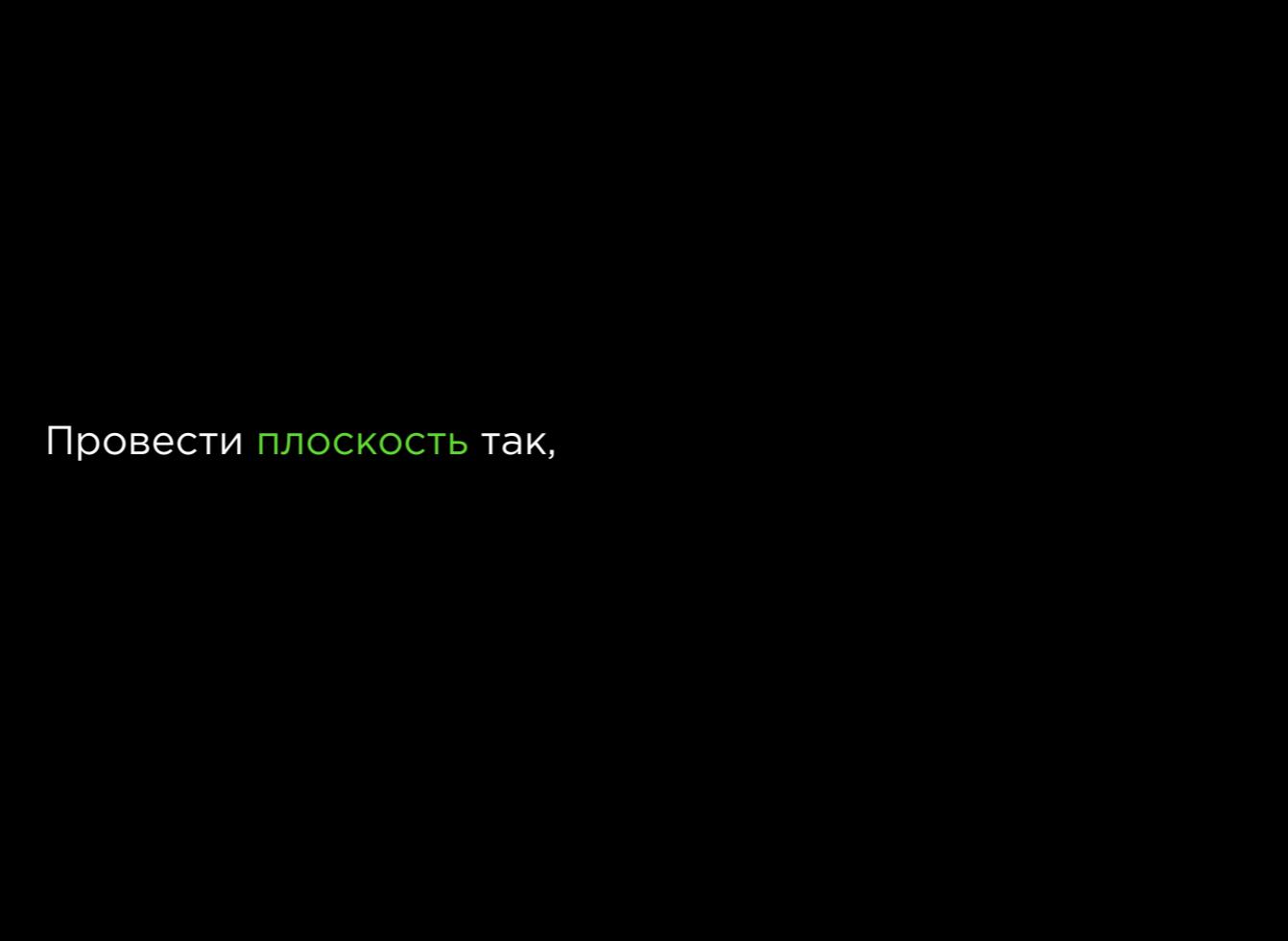
В многомерном случае у нас была бы такая плоскость, которую мы бы построили используя эти точки.

Ну и так далее по размерностям.



Т.е. задача звучит так провести плоскость, используя точки в пространстве так, чтобы все точки были как можно ближе к плоскости.

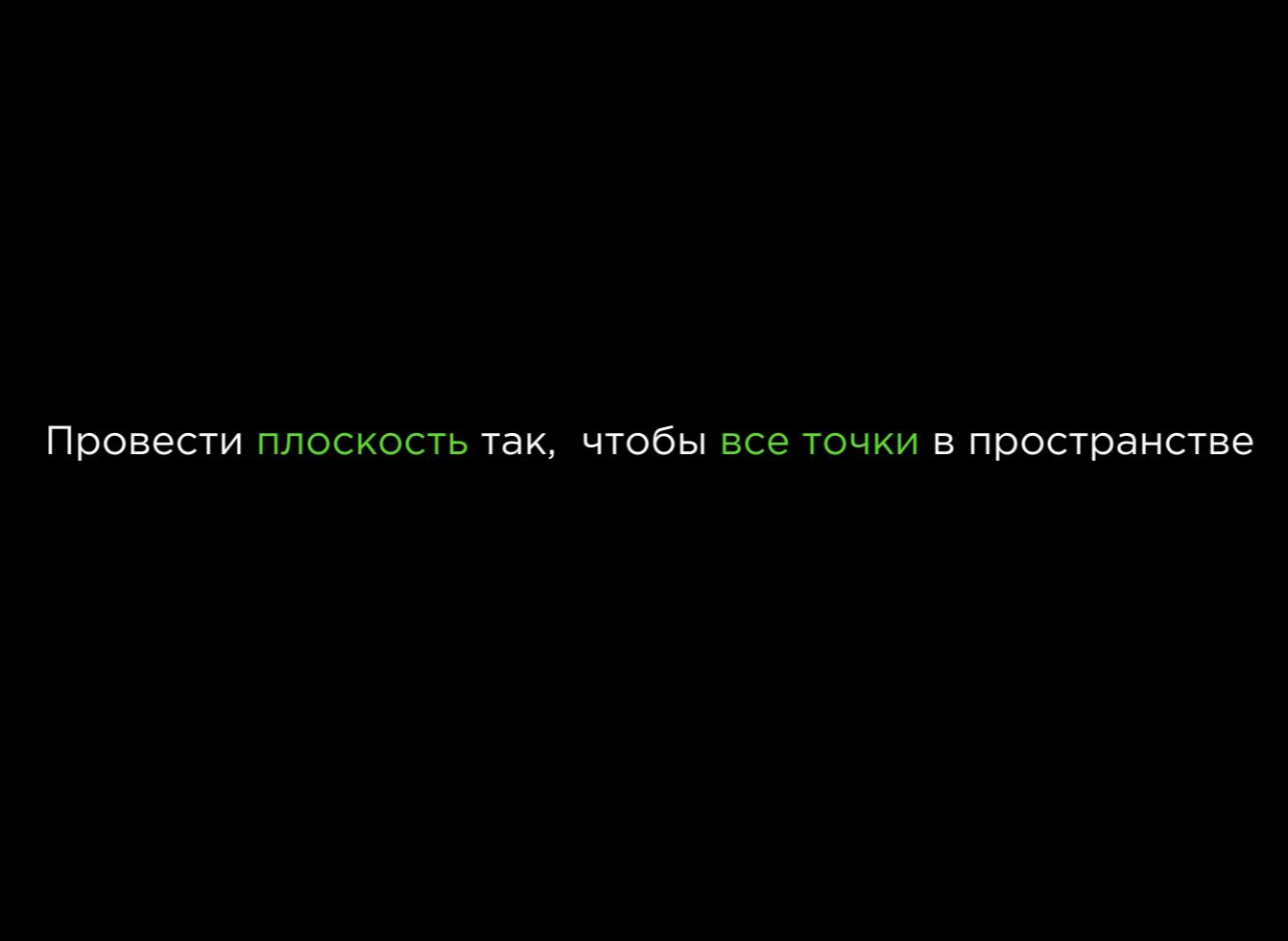
Возможно, кому-то это все еще не очень формальная формулировка, однако тут можно заметить слова “Как можно ближе” “точки” и “плоскость”



Провести **плоскость** так,

Т.е. задача звучит так провести плоскость, используя точки в пространстве так, чтобы все точки были как можно ближе к плоскости.

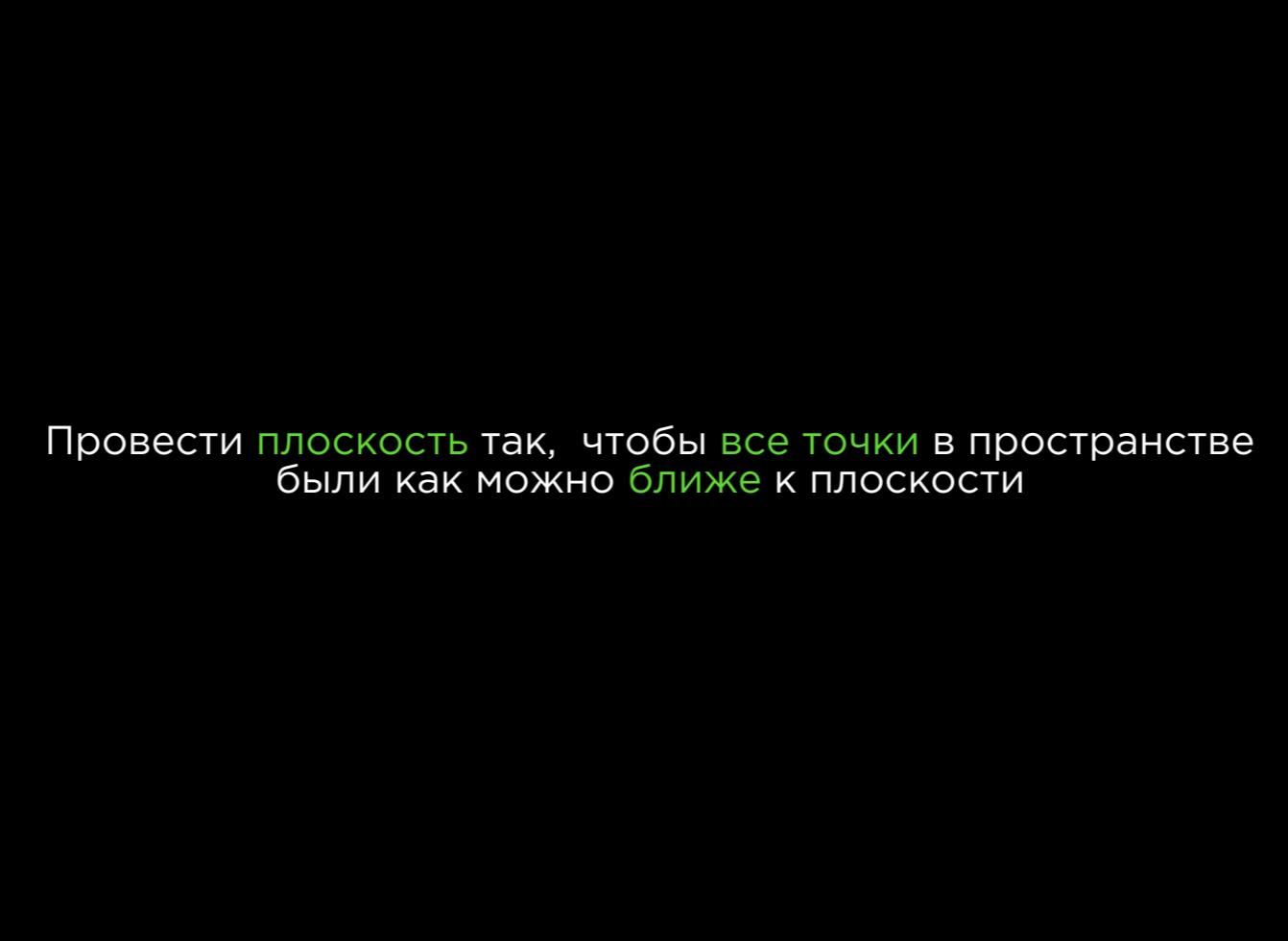
Возможно, кому-то это все еще не очень формальная формулировка, однако тут можно заметить слова “Как можно ближе” “точки” и “плоскость”



Провести **плоскость** так, чтобы **все точки** в пространстве

Т.е. задача звучит так провести плоскость, используя точки в пространстве так, чтобы все точки были как можно ближе к плоскости.

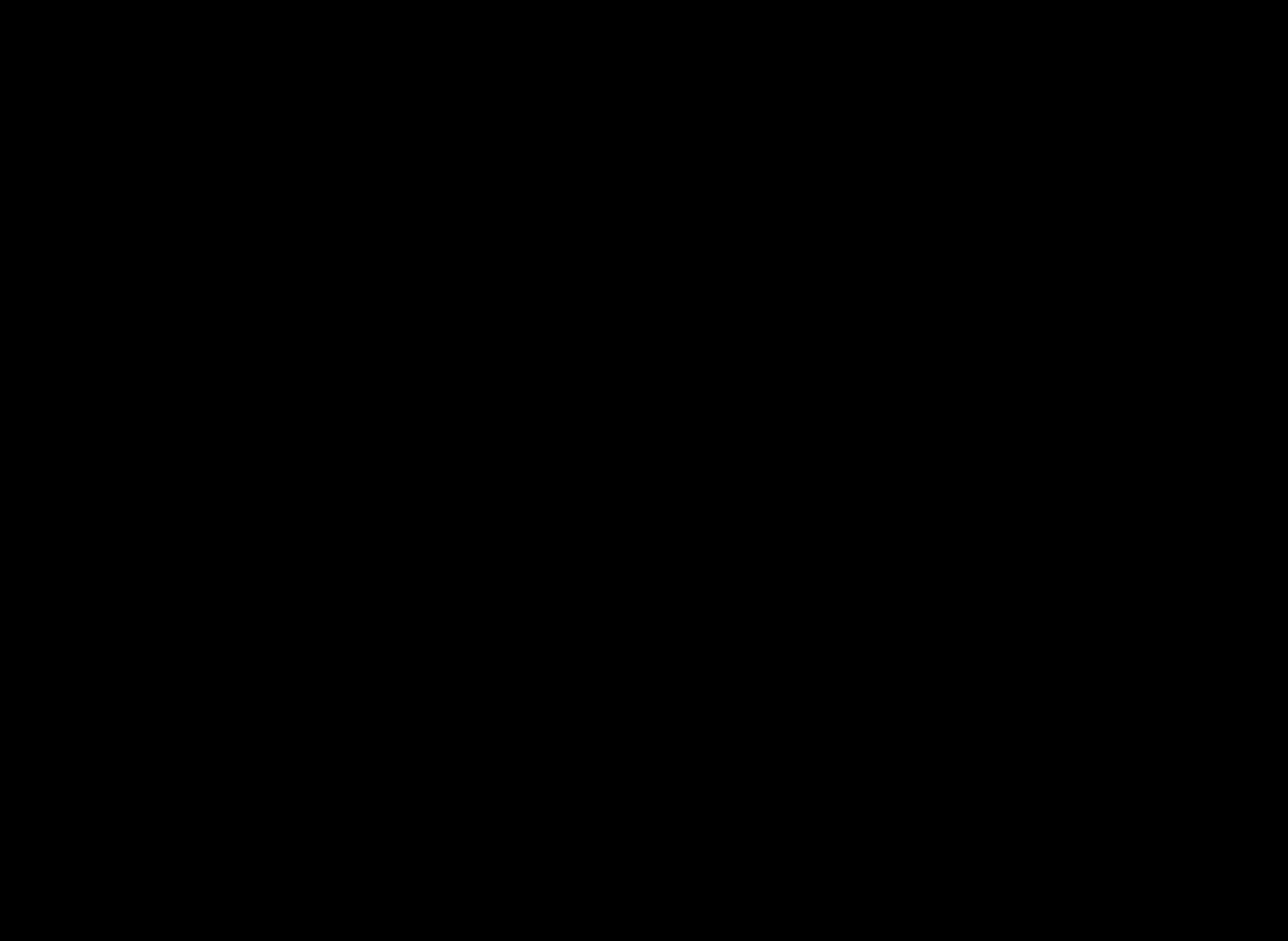
Возможно, кому-то это все еще не очень формальная формулировка, однако тут можно заметить слова “Как можно ближе” “точки” и “плоскость”



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

Т.е. задача звучит так провести плоскость, используя точки в пространстве так, чтобы все точки были как можно ближе к плоскости.

Возможно, кому-то это все еще не очень формальная формулировка, однако тут можно заметить слова “Как можно ближе” “точки” и “плоскость”



Во-первых, мы знаем, как выглядит уравнение плоскости
Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным
Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b .

$$Y = Xa + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

$$Y = Xa + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

$$Y = Xa + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

Хотим найти

$$Y = Xa + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

Хотим найти

$$Y = Xa + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

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

Хотим найти

$$Y = Xa + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

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

Хотим найти

$$Y = Xa + b$$

$$a_1x_{1,1} + a_2x_{1,2} + a_3x_{1,3} + \dots + a_nx_{1,n} + b$$

$$a_1x_{2,1} + a_2x_{2,2} + a_3x_{2,3} + \dots + a_nx_{2,n} + b$$

$$\vdots$$

$$a_1x_{l,1} + a_2x_{l,2} + a_3x_{l,3} + \dots + a_nx_{l,n} + b$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

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

Хотим найти

$$Y = Xa + b$$

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

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

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

Хотим найти

$$Y = Xa + b$$

$$\begin{array}{rcl} a_1x_{1,1} + a_2x_{1,2} + a_3x_{1,3} + \dots + a_nx_{1,n} + b & & +1 * b \\ a_1x_{2,1} + a_2x_{2,2} + a_3x_{2,3} + \dots + a_nx_{2,n} + b & \longrightarrow & \dots +1 * b \\ \vdots \\ a_1x_{l,1} + a_2x_{l,2} + a_3x_{l,3} + \dots + a_nx_{l,n} + b & & +1 * b \end{array}$$

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

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

Хотим найти

$$Y = X'w$$

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

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

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

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

Хотим найти

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

$$Y = X'w$$

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

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.



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

Во-первых, мы знаем, как выглядит уравнение плоскости

Во-вторых, Нам известны Y и x

И нам нужны A и b по этим данным

Мы также можем все это написать в таком виде, представив b в виде единичного вектора на b.

$$Y = X'w$$

Получается, у нас есть система уравнений

И хочется сказать, “А теперь просто возьмем обратную матрицу от X и получим искомый ответ”

Но нет

Потому мы можем взять обратную матрицу от квадратной

Так что привет приближенный метод

$$Y = X'w$$

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

Получается, у нас есть система уравнений

И хочется сказать, “А теперь просто возьмем обратную матрицу от X и получим искомый ответ”

Но нет

Потому мы можем взять обратную матрицу от квадратной

Так что привет приближенный метод

$$Y = X'w$$

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

Получается, у нас есть система уравнений

И хочется сказать, “А теперь просто возьмем обратную матрицу от X и получим искомый ответ”

Но нет

Потому мы можем взять обратную матрицу от квадратной

Так что привет приближенный метод

Метод наименьших квадратов

Выводим решение для коэффициентов

Формулировка МНК

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

Задача звучит так занудно, но, по факту это просто разница суммы квадратов разницы

$$Xw = Y$$

И первый вариант решения

Это будет вполне себе решение задачи регрессии. Но этот метод не будет всегда работать

$$\begin{aligned} Xw &= Y \\ X^T Xw &= X^T Y \end{aligned}$$

И первый вариант решения

Это будет вполне себе решение задачи регрессии. Но этот метод не будет всегда работать

$$\begin{aligned}Xw &= Y \\ X^T X w &= X^T Y \\ w &= (X^T X)^{-1} X^T Y\end{aligned}$$

И первый вариант решения

Это будет вполне себе решение задачи регрессии. Но этот метод не будет всегда работать

Проблемы

Проблемы связанные с аналитическим решением

Проблемы

- Имеется операция взятия обратной матрицы

Проблемы связанные с аналитическим решением

Проблемы

$O(n^3)$ операций

- Имеется операция взятия обратной матрицы

Проблемы связанные с аналитическим решением

Проблемы

$O(n^3)$ операций

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

Проблемы связанные с аналитическим решением

Проблемы

$O(n^3)$ операций

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

Тогда матрица
необратима

Проблемы связанные с аналитическим решением

Что мы делаем, когда
невозможно аналитически?



Апроксимируем!

Градиентный спуск

Градиентный спуск

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

Эта штука эквивалентна такой
Нам это пригодится

Градиентный спуск

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

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

Эта штука эквивалентна такой
Нам это пригодится

Градиентный спуск

$$Q(w, X)$$

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

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

Эта штука эквивалентна такой
Нам это пригодится

Градиентный спуск

Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Градиентный спуск

$$w^0 = 0$$

Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Градиентный спуск

$$w^0 = 0$$

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

Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

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 — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Градиентный спуск



Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Градиентный спуск



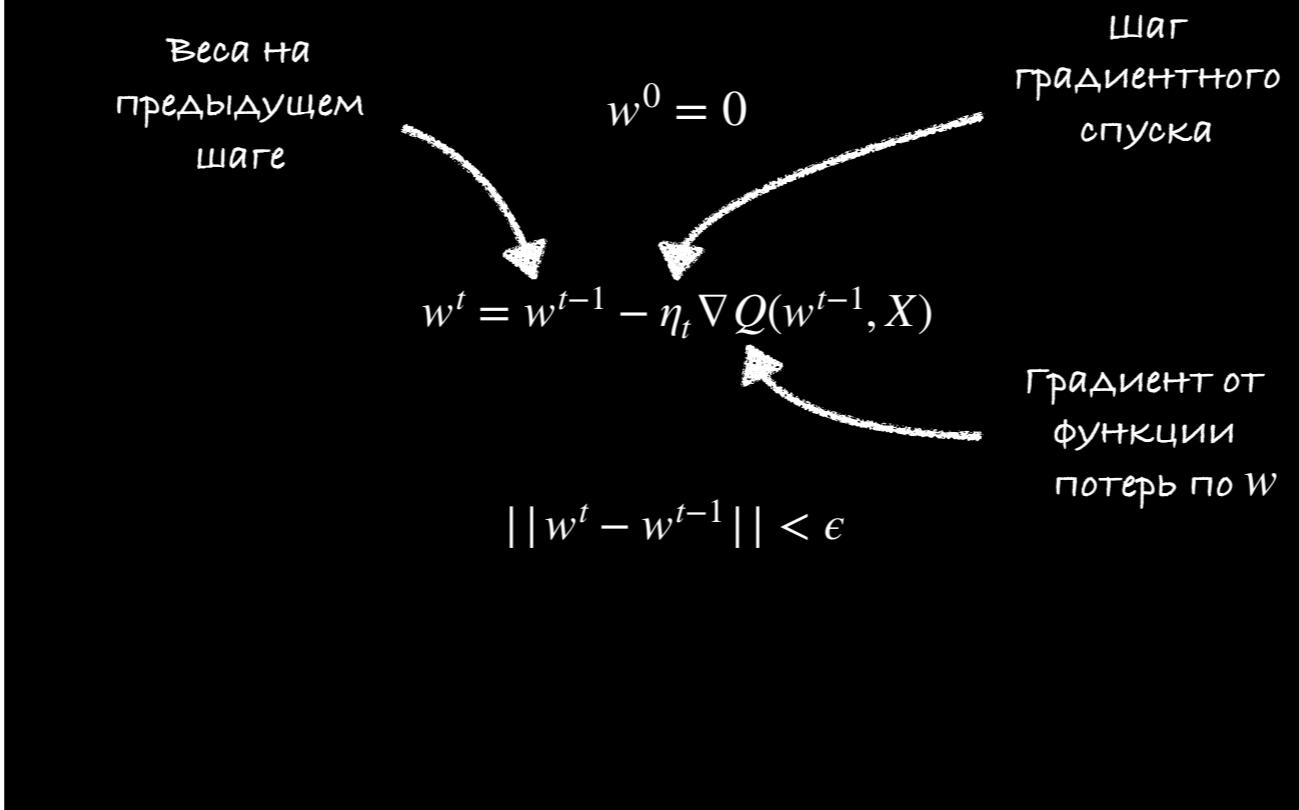
Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Градиентный спуск



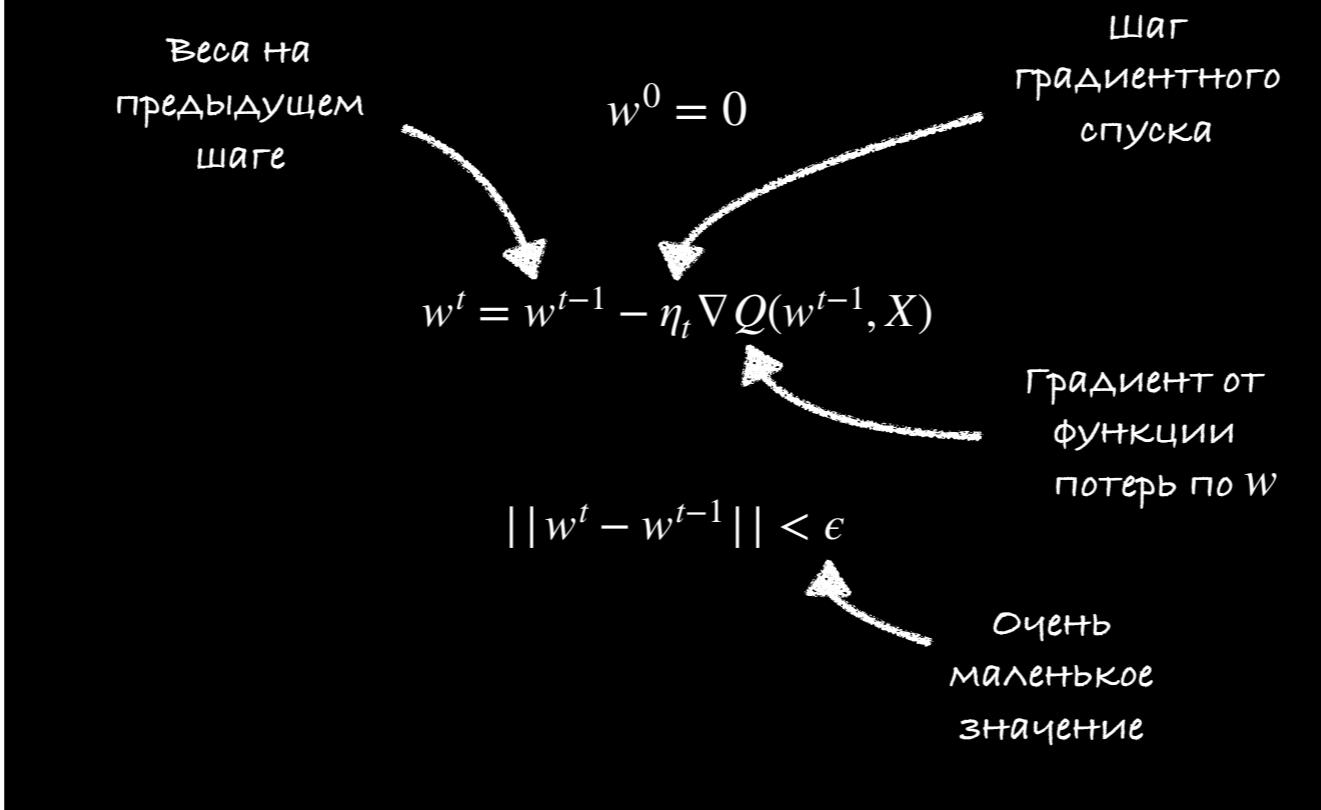
Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Градиентный спуск



Градиентный спуск — это итерационный метод

Так что ему нужно дать что-то сначала

Это будет вектор нулевых весов

Затем мы будем обновлять эти веса по правилу 2, пока не дойдем до какого-то очень маленького значений эпсилон

Комментарий насчет шага

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

Выбор шага градиентного спуска — это отдельная проблема. Если много способов, но на данный момент, можете думать, что этот шаг равен некоторой константе на номер итерации.

Комментарий насчет шага

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

Выбор шага градиентного спуска — это отдельная проблема. Если много способов, но на данный момент, можете думать, что этот шаг равен некоторой константе на номер итерации.

Комментарий насчет шага

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

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

Выбор шага градиентного спуска — это отдельная проблема. Если много способов, но на данный момент, можете думать, что этот шаг равен некоторой константе на номер итерации.

Комментарий насчет шага

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

константа
(сами выбираем)

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

Выбор шага градиентного спуска — это отдельная проблема. Если много способов, но на данный момент, можете думать, что этот шаг равен некоторой константе на номер итерации.

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

Выбор шага градиентного спуска — это отдельная проблема. Если много способов, но на данный момент, можете думать, что этот шаг равен некоторой константе на номер итерации.

Как посчитать градиент?

Вопрос только в том, как же посчитать градиент?

Like baby steps

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

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

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

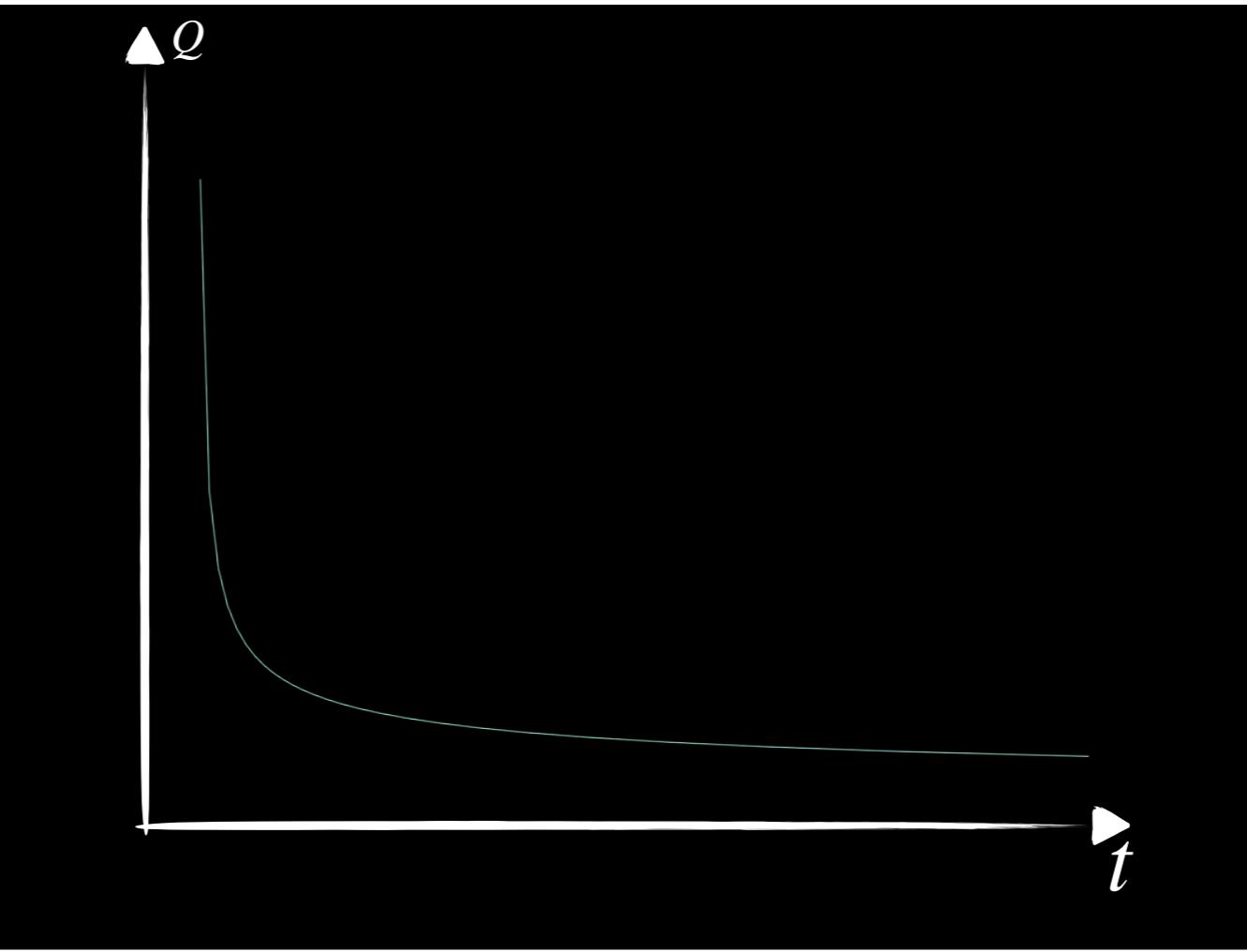
$$\frac{\partial Q}{\partial w_0} = \frac{2}{l} \sum_{i=1}^l (w_1 x_i + w_0 - y_i)$$

Начнем с двухмерного случая

Обобщим

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

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



Так будет выглядеть функция потерь после определенного количества итераций

Проблемы

- Он может долго считаться при большой выборке

Проблемы, связанные с градиентным спуском

Суть в том, что если у нас очень много примеров I , то будет происходить очень много перемножений матрицы X на w .

Поэтому, часто применяют другой метод

Проблемы

- Он может долго считаться при большой выборке

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

Проблемы, связанные с градиентным спуском

Суть в том, что если у нас очень много примеров l , то будет происходить очень много перемножений матрицы X на w .

Поэтому, часто применяют другой метод

Проблемы

- Он может долго считаться при большой выборке

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

Примеров может
быть очень много

Проблемы, связанные с градиентным спуском

Суть в том, что если у нас очень много примеров l , то будет происходить очень много перемножений матрицы X на w .

Поэтому, часто применяют другой метод

Стохастический градиентный спуск

И проблему до

Стохастический градиентный спуск

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



Рассказать суть метода

Что идем и батчам и, в простейшем случае, идем по одним наблюдениям

В этом и вся модификация SGD

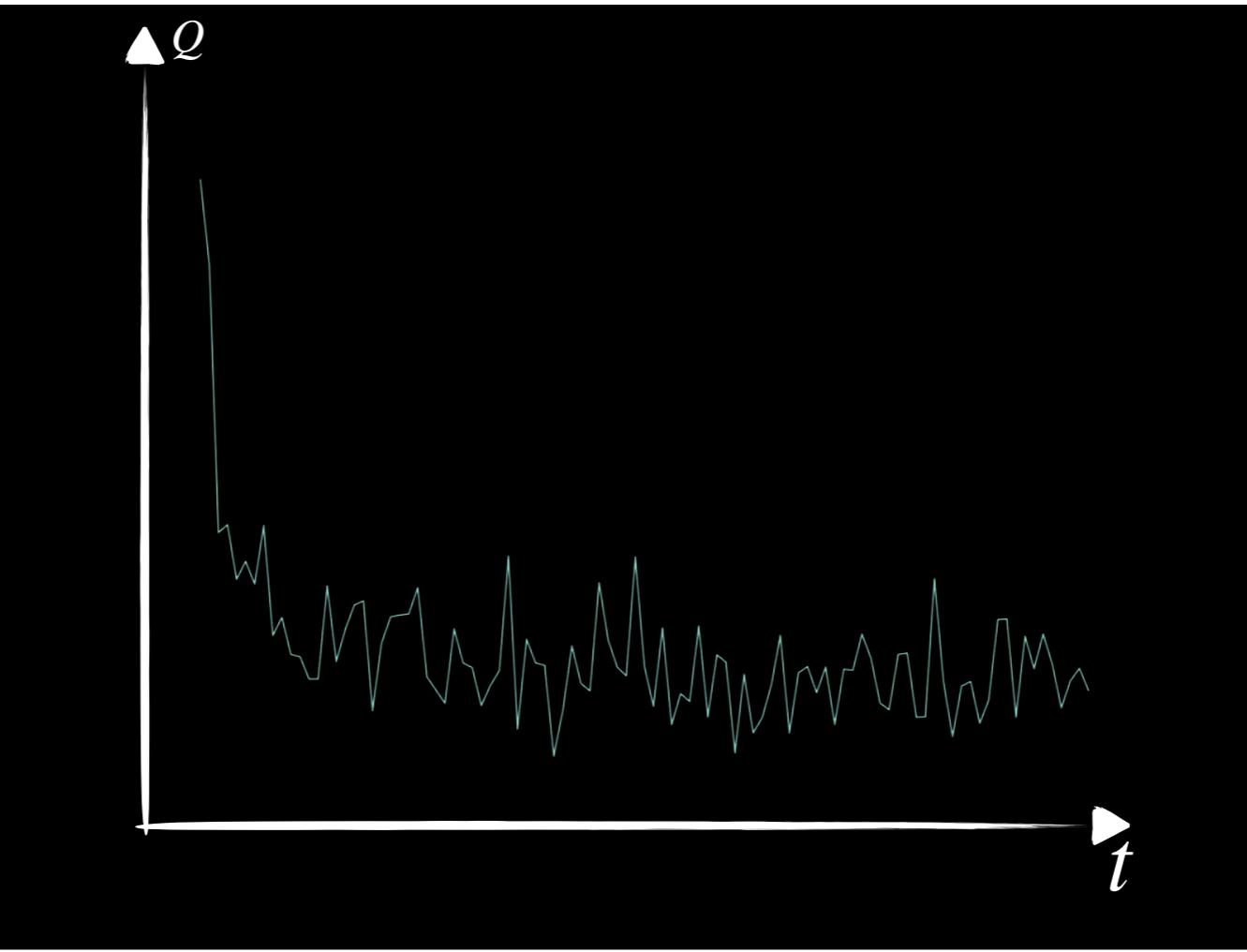
Стохастический градиентный спуск



Рассказать суть метода

Что идем и батчам и, в простейшем случае, идем по одним наблюдениям

В этом и вся модификация SGD



Тогда функция потерь будет выглядеть чуть более ломаной, но все равно, алгоритм будет сходиться.

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

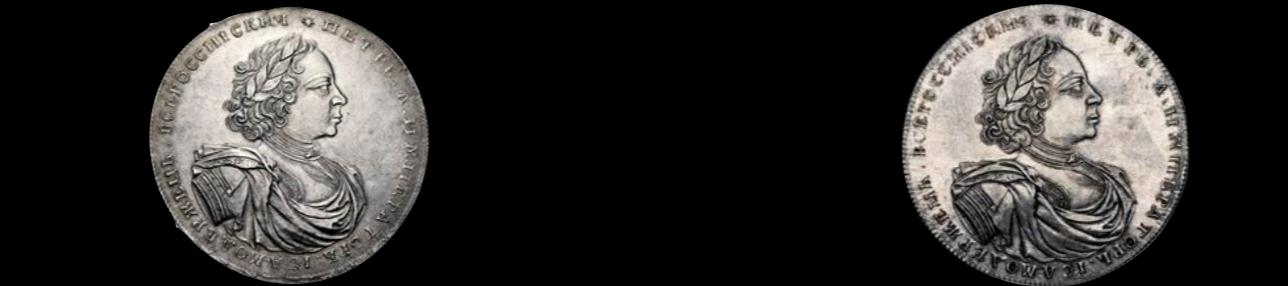
~~$Y \in \mathbb{R}$~~ Регрессия

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

Бинарная классификация

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

Пример



Примером бинарной классификации может быть любой пример, когда есть две разных сущности, которые надо различить: Что-то подлинное и ложное или одно отличить от другого.

Пример

1



0



Примером бинарной классификации может быть любой пример, когда есть две разных сущности, которые надо различить: Что-то подлинное и ложное или одно отличить от другого.

Пример

1

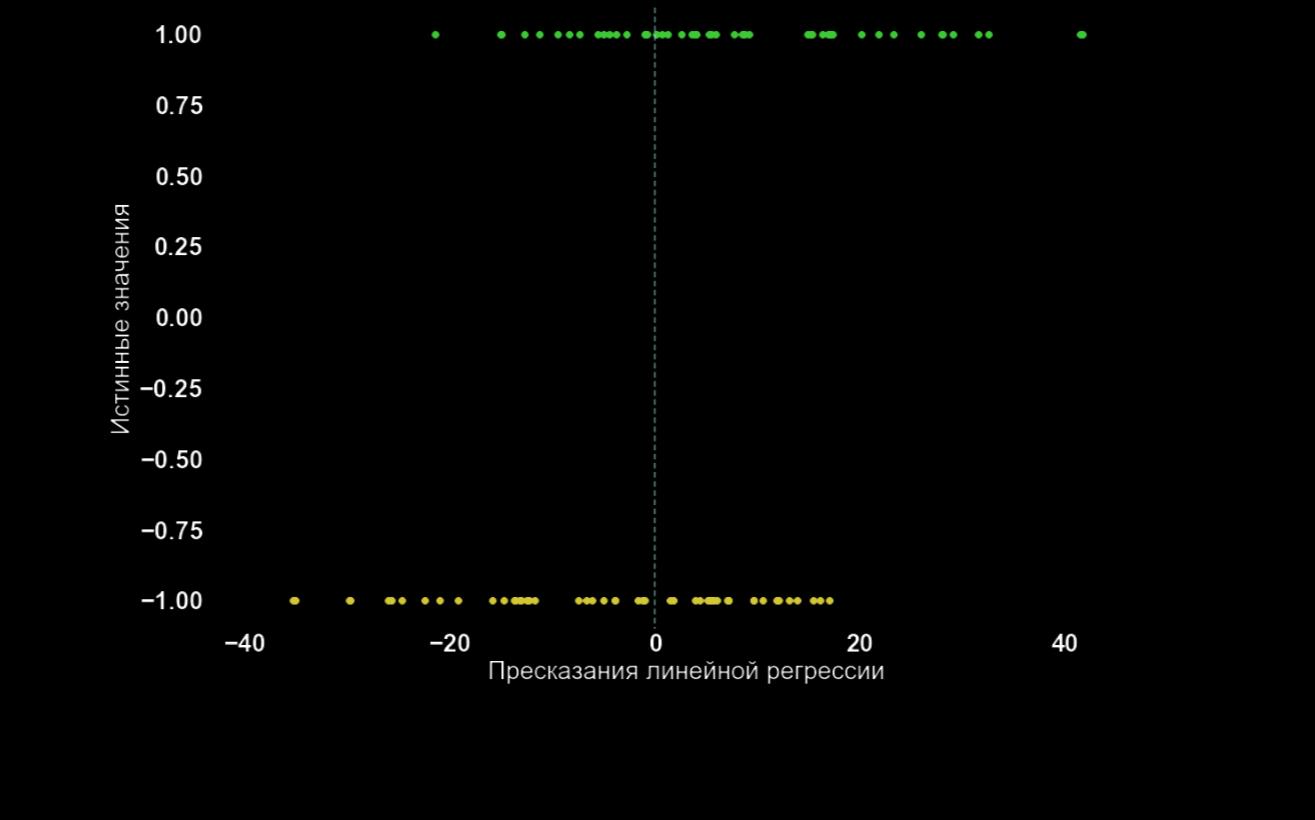


0



Примером бинарной классификации может быть любой пример, когда есть две разных сущности, которые надо различить: Что-то подлинное и ложное или одно отличить от другого.

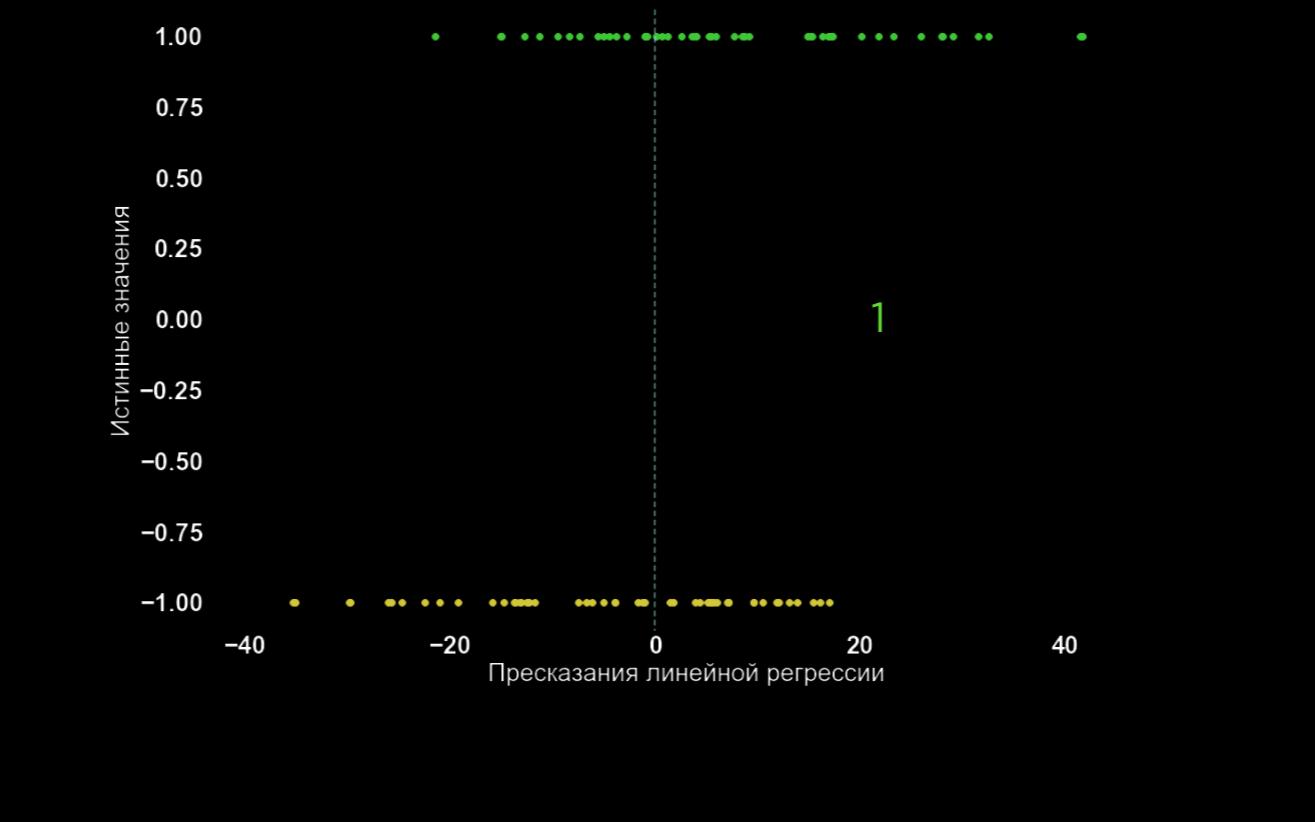
Регрессия для классификации



Если мы попробуем просто использовать личную регрессию для классификации, получится что-то вроде такого. И мы можем отсечь предсказания линейной модели по нулю, сказав, что все, что меньше нуля — это -1, а больше нуля — 1

И есть функция, которая это делает — sign

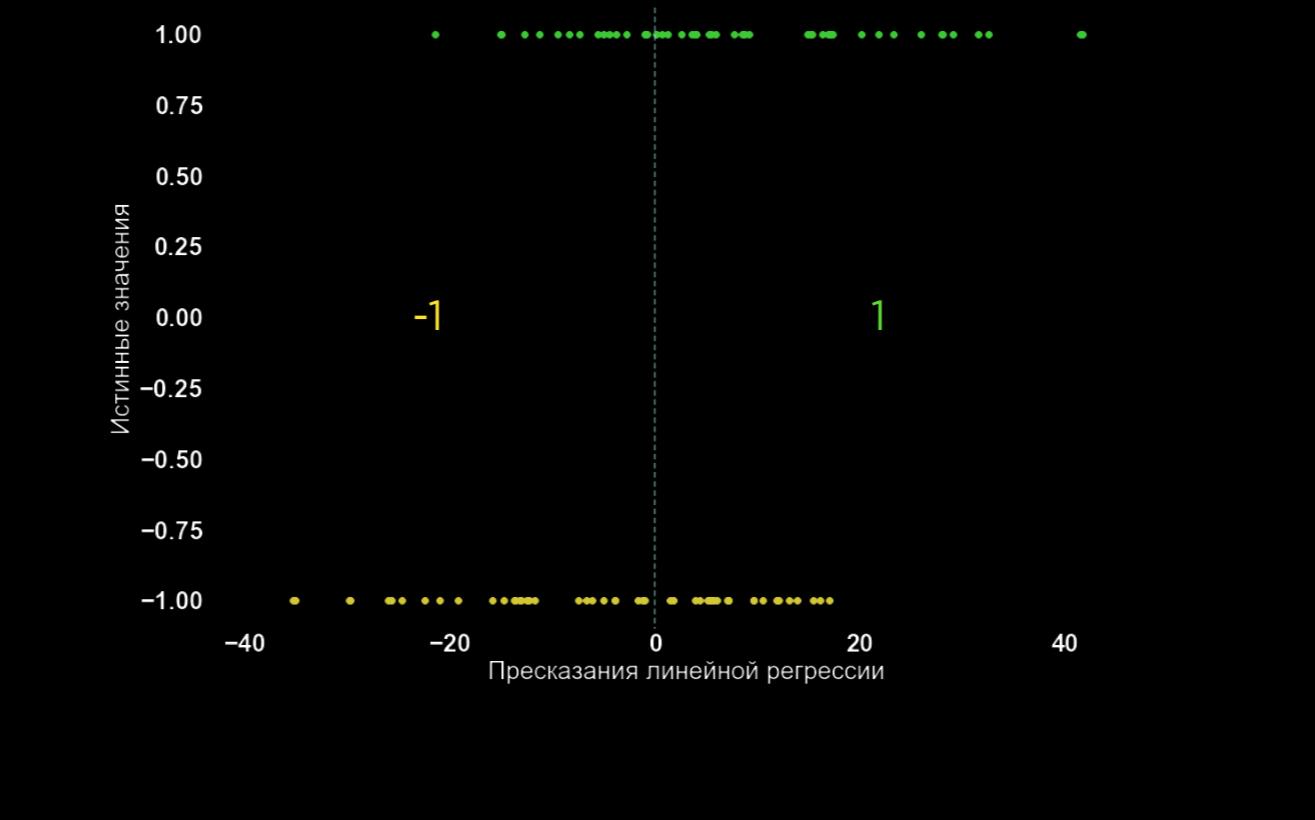
Регрессия для классификации



Если мы попробуем просто использовать личную регрессию для классификации, получится что-то вроде такого. И мы можем отсечь предсказания линейной модели по нулю, сказав, что все, что меньше нуля — это -1, а больше нуля — 1

И есть функция, которая это делает — sign

Регрессия для классификации



Если мы попробуем просто использовать личную регрессию для классификации, получится что-то вроде такого. И мы можем отсечь предсказания линейной модели по нулю, сказав, что все, что меньше нуля — это -1, а больше нуля — 1

И есть функция, которая это делает — sign

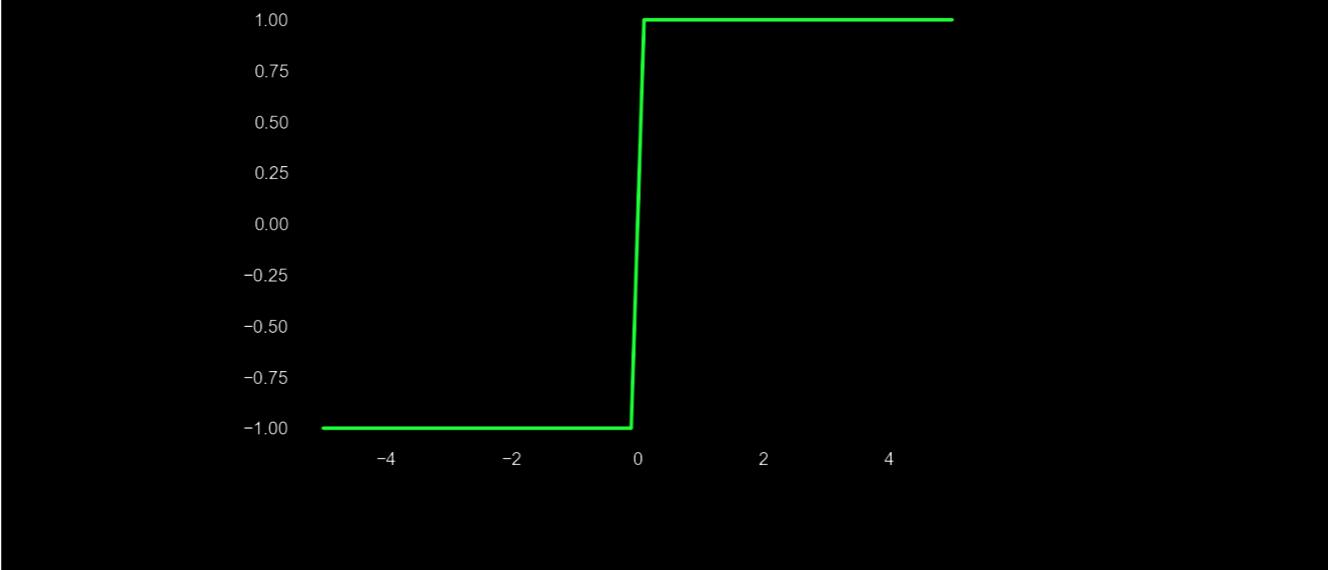
От регрессии к классификации

$$Y = Xw + b$$

И эта функция будет принимать решение регрессии, а отдавать знак -1 или 1

От регрессии к классификации

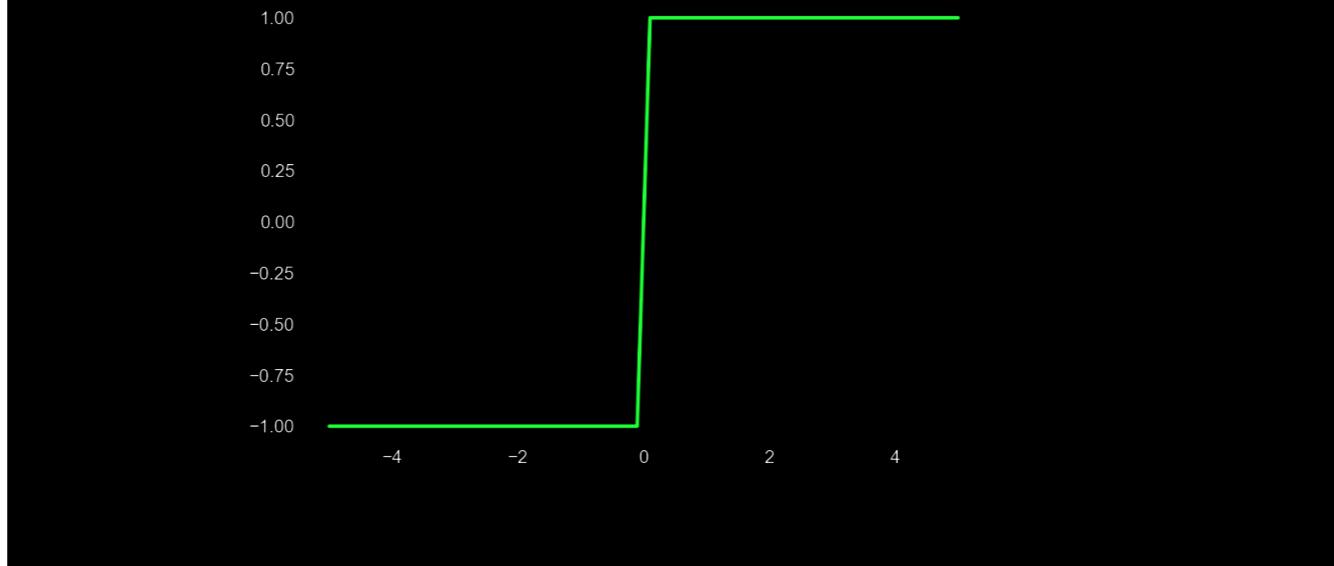
$$Y = \text{sign}(Xw + b)$$



И эта функция будет принимать решение регрессии, а отдавать знак -1 или 1

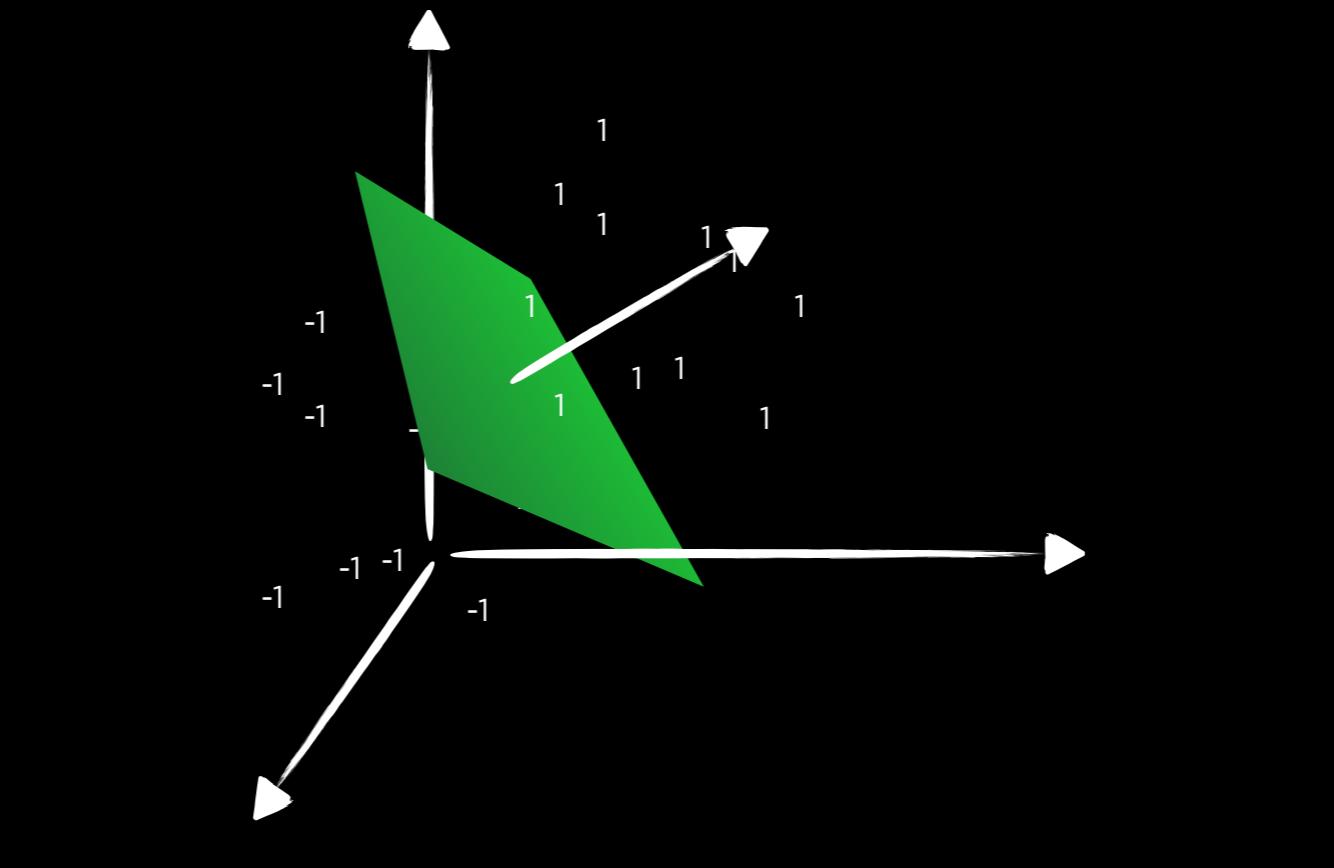
От регрессии к классификации

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



И эта функция будет принимать решение регрессии, а отдавать знак -1 или 1

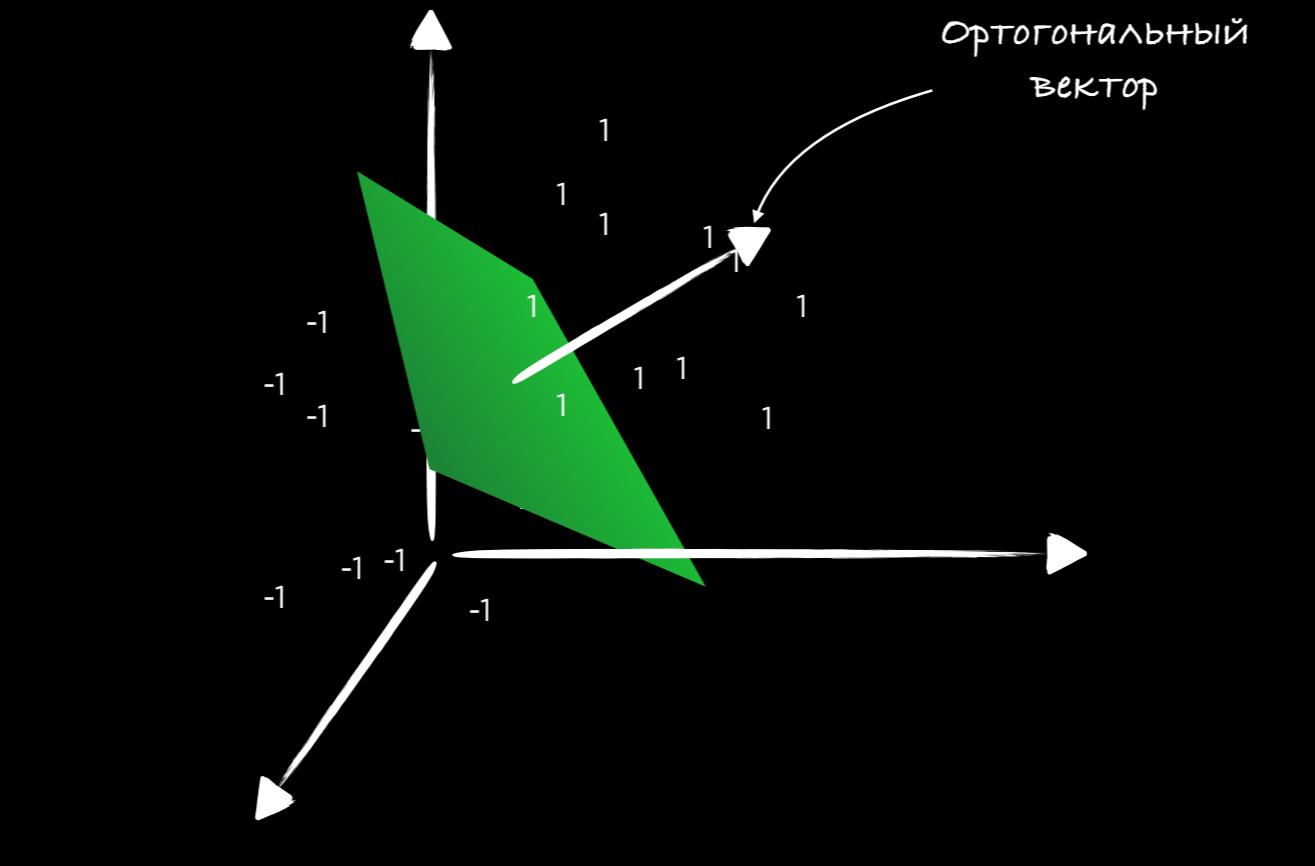
Почему?



Почему это работает?

Потому что все наши точки — это -1 и 1. А Веса модели, на самом деле, определяют нормальный вектор к плоскости. Потому оно и будет работать.

Почему?



Почему это работает?

Потому что все наши точки — это -1 и 1. А Веса модели, на самом деле, определяют нормальный вектор к плоскости. Потому оно и будет работать.

Но

- Хотим использовать методы оптимизации, которые выше

Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

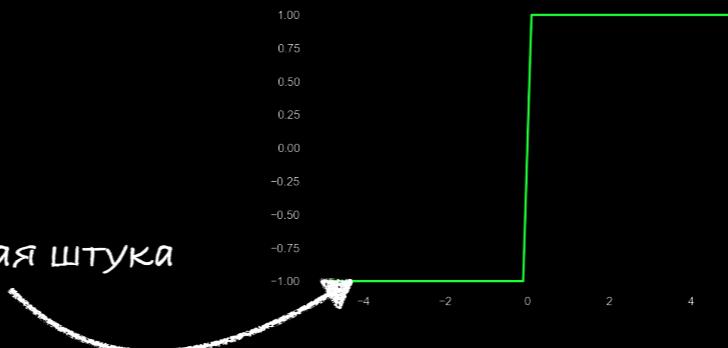
Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

Но

- Хотим использовать методы оптимизации, которые выше

Нелинейная штука



Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

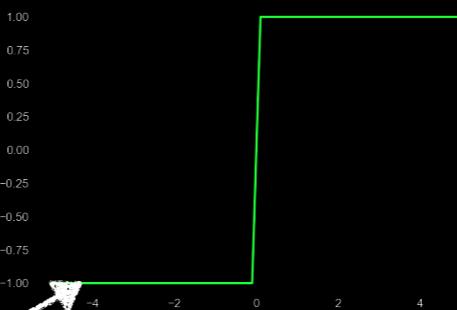
Но

- Хотим использовать методы оптимизации, которые выше

А нам нужны
производные



Нелинейная штука



Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

Но

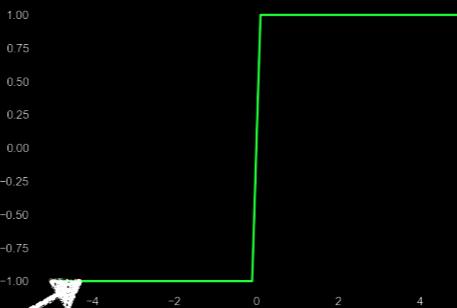
$$Y = f(X_1, \dots, X_l) + \epsilon$$

- Хотим использовать методы оптимизации, которые выше

А нам нужны
производные



Нелинейная штука



Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

Но

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

Раньше

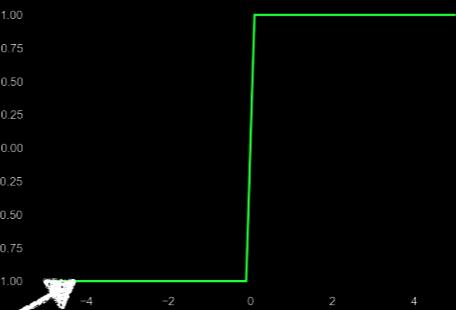
$$Xw + b$$

- Хотим использовать методы оптимизации, которые выше

А нам нужны
производные



Нелинейная штука



Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

Но

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

раньше

теперь

$Xw + b$

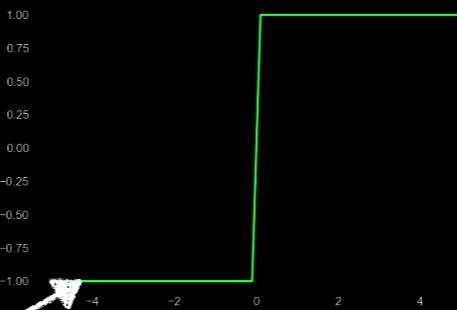
$sign(Xw + b)$

- Хотим использовать методы оптимизации, которые выше

А нам нужны
производные



Нелинейная штука



Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

Но

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

теперь

$sign(Xw + b)$

раньше

$Xw + b$

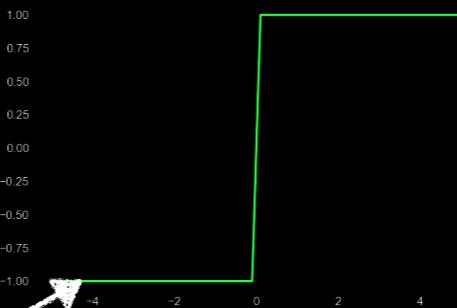
∇Q

- Хотим использовать методы оптимизации, которые выше

А нам нужны
производные



Нелинейная штука

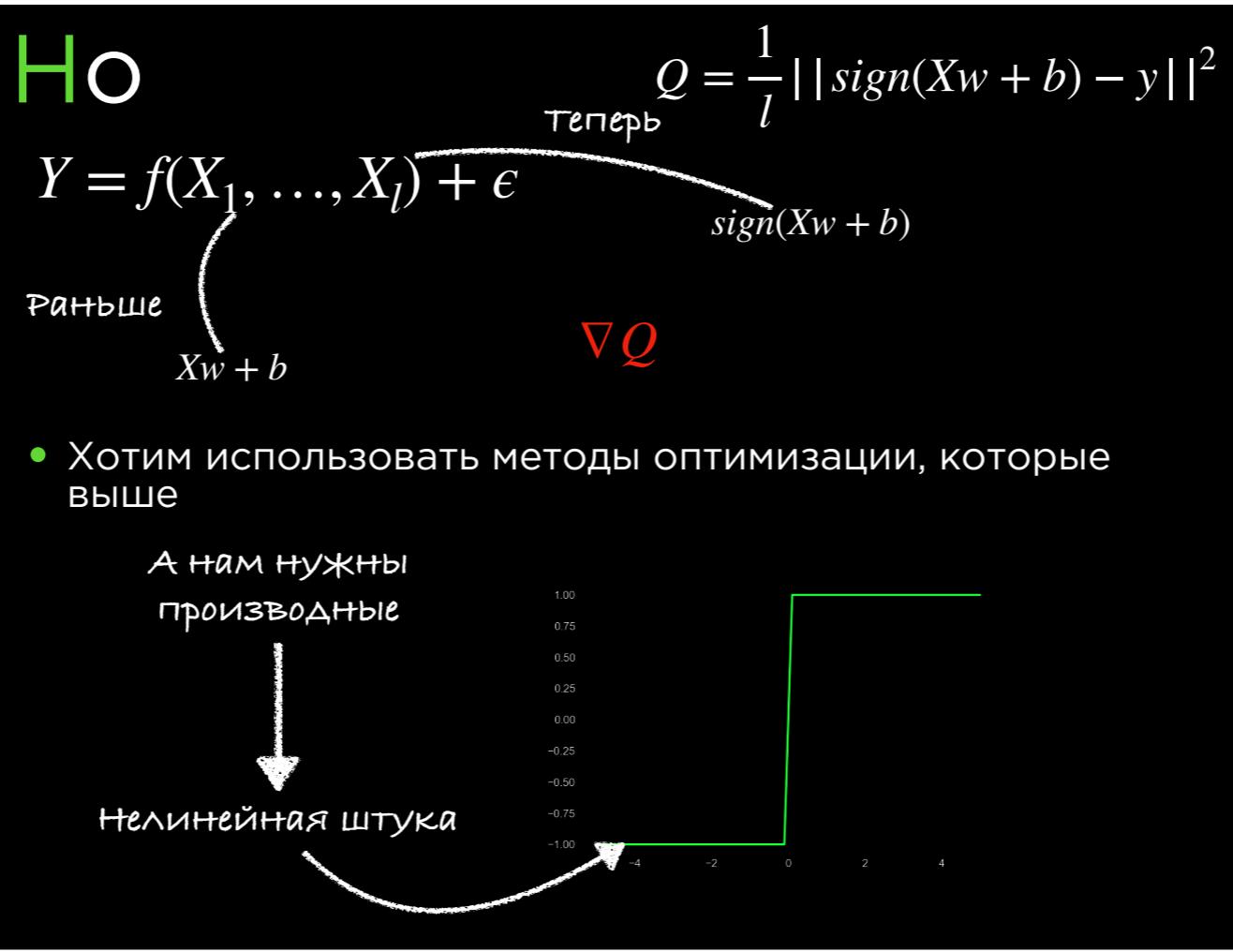


Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

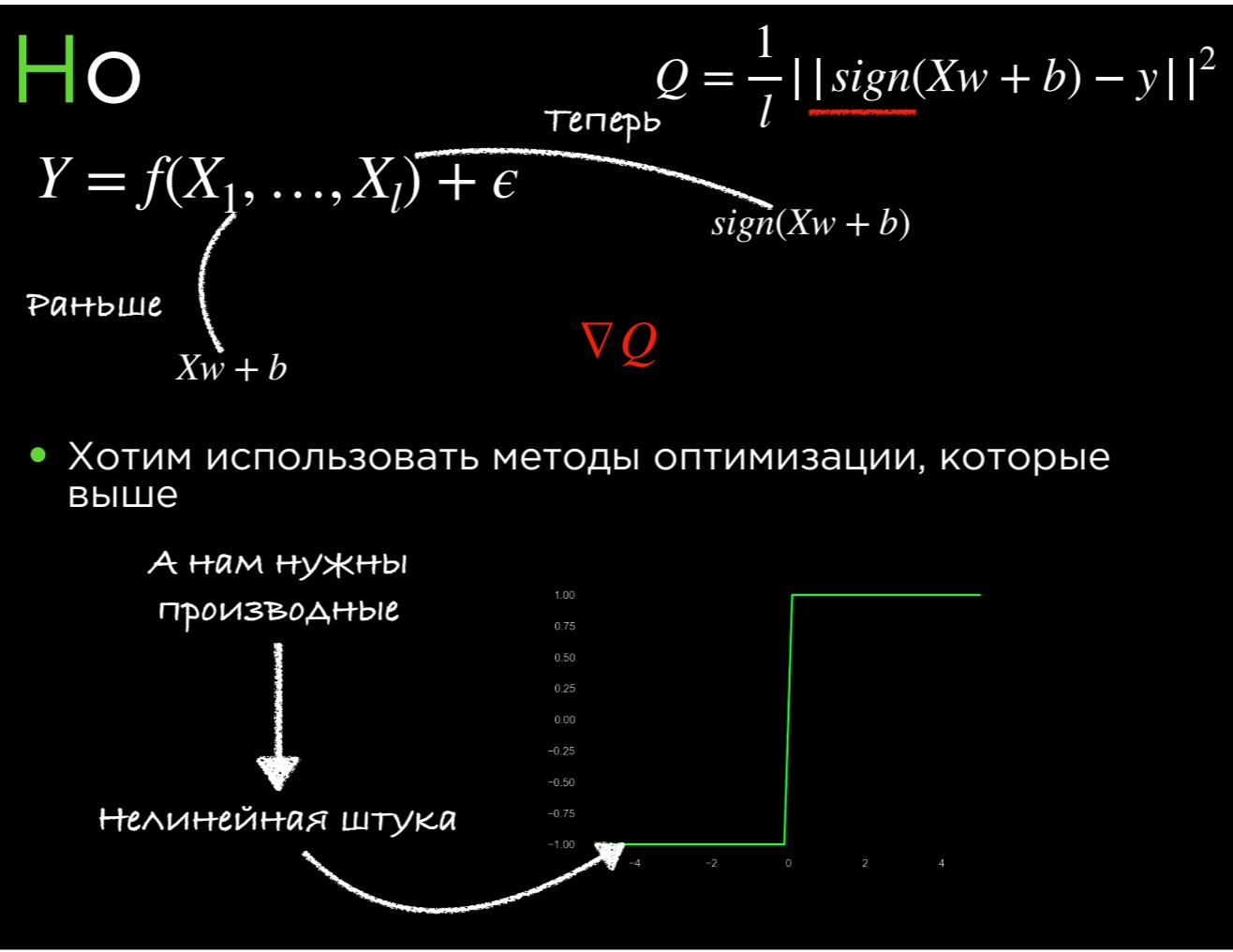


Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции



Но мы хотим использовать те методы оптимизации, которые описаны выше.

Но у нас есть небольшие проблемы с нелинейностью функции, которую мы считаем.

Теперь градиент функции потерь не так просто считается, как при $Xw+b$

Теперь надо еще брать производную по функции активации. А мы не можем брать производную от этой функции

Логистическая регрессия

И потому, нас может спасти другая функция активации.

Логистическая регрессия

Logistic for logits

Logits

$$p = \text{sigmoid}(Xw + b)$$

Сигмойда))

Что за сигмойда?

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

Почти как линейный классификатор, только теперь активация гладкая и функция возвращает значения от 0 до 1.

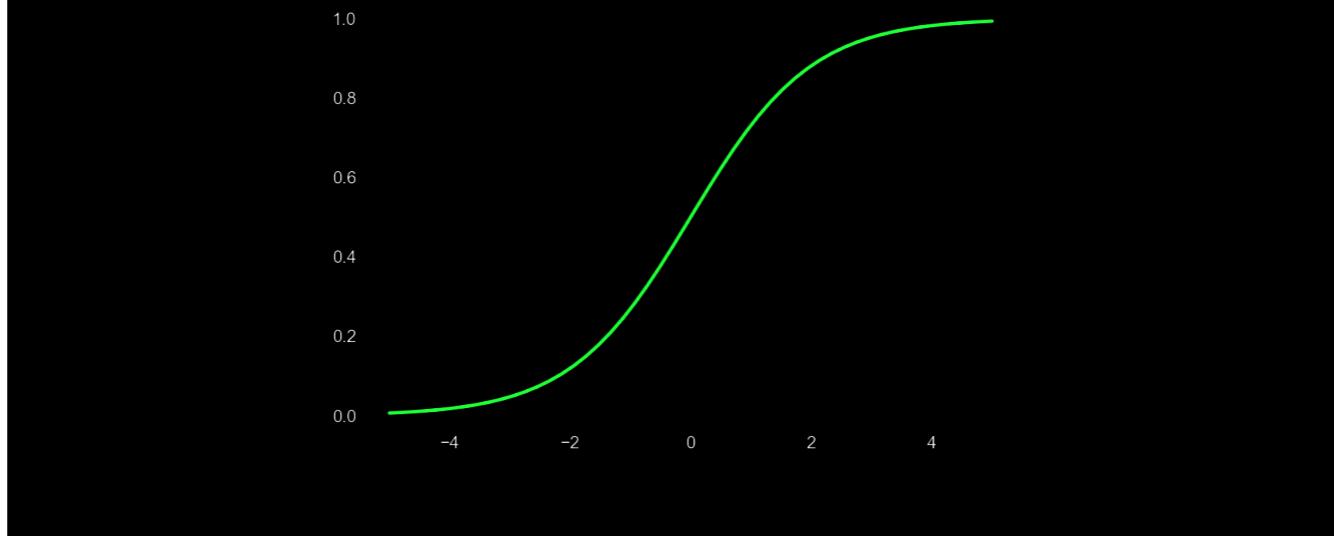
Что похоже на что-то около вероятности.

Т.е. Если в линейном классификаторе мы имели дело с регрессией, закрученной изолентой, то теперь у нас в распоряжении что-то похожее на вероятности.
А также **гладкое**.

Это будет полезно дальше.

Что за сигмойда?

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



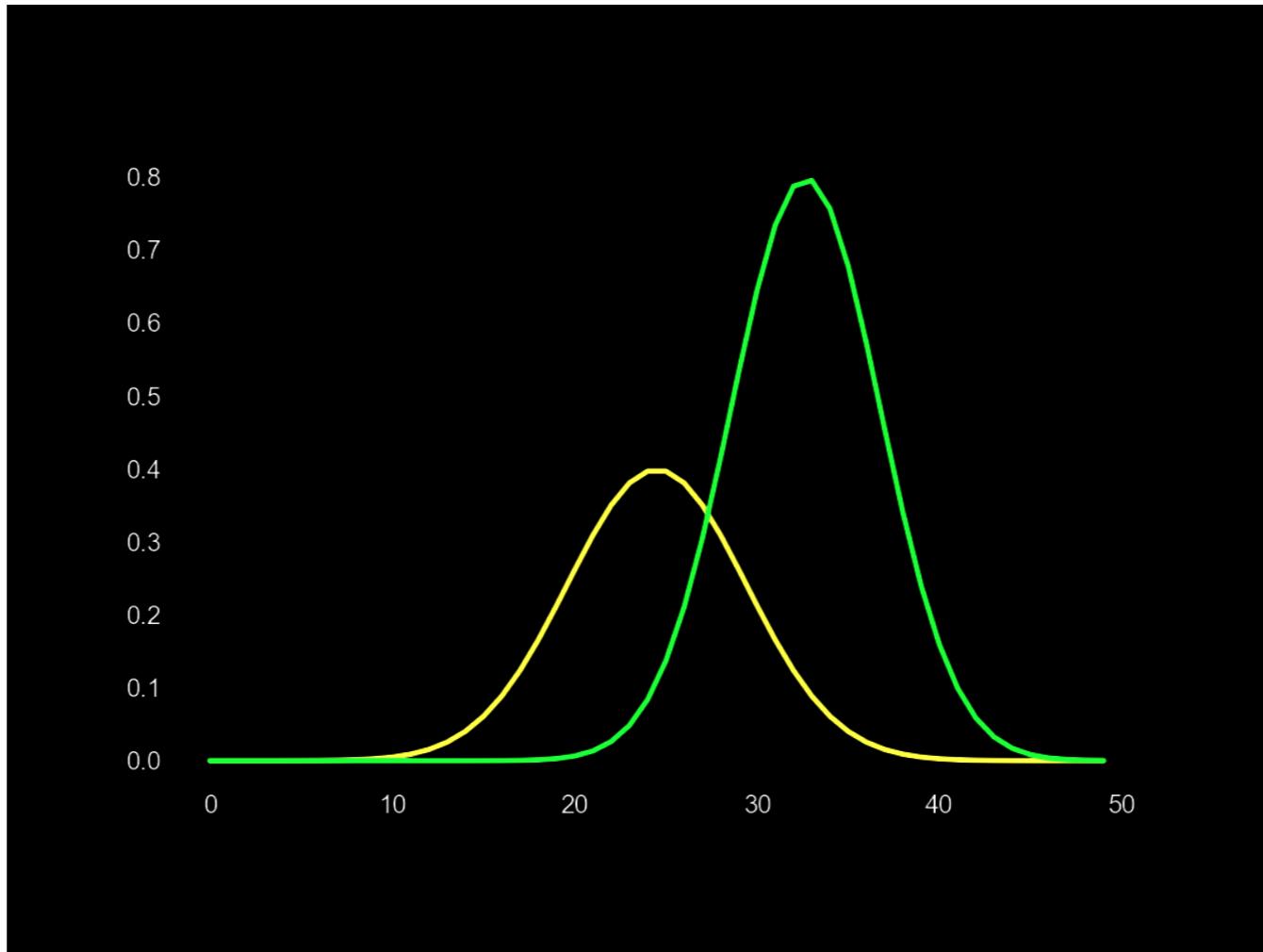
Почти как линейный классификатор, только теперь активация гладкая и функция возвращает значения от 0 до 1.

Что похоже на что-то около вероятности.

Т.е. Если в линейном классификаторе мы имели дело с регрессией, закрученной изолентой, то теперь у нас в распоряжении что-то похожее на вероятности.
А также **гладкое**.

Это будет полезно дальше.

Функция потерь



И так, в логистической регрессии, вообще говоря, мы теперь минимизируем расстояние между двумя распределениями вероятностей. И для такого расстояния используется другая мера расстояния.

Cross entropy

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

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

$$y_i = 1$$

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

$$y_i = 1$$

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

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

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

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

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

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

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

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

$$y_i = 1$$

$$y_i = 0$$

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

p_i Как можно ближе к 1

Мало если правильно

Велико если неправильно

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

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

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

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

p_i Как можно ближе к 1

Мало если правильно

Велико если неправильно

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

$$y_i = 1$$

$$y_i = 0$$

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

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

p_i Как можно ближе к 1

Мало если правильно

Велико если неправильно

p_i Как можно ближе к 0

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

$$y_i = 1$$

$$y_i = 0$$

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

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

p_i Как можно ближе к 1

p_i Как можно ближе к 0

Мало если правильно

Мало если правильно

Велико если неправильно

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Cross entropy

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

$$y_i = 1$$

$$y_i = 0$$

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

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

p_i Как можно ближе к 1

Мало если правильно

Велико если неправильно

p_i Как можно ближе к 0

Мало если правильно

Велико если неправильно

Почему именно эта функция потерь?

Потому что мы, вообще говоря, хотим, чтобы мы получали как можно меньшее значение в случае правильных ответов.

Градиент

Посмотрим теперь на градиент от функции потерь.

ФУНКЦИИ ОШИБКИ

Такая функция потерь была раньше, и мы разобрались, как брать от нее производную.

Переть посмотрим, как брать производную от этой штуки, потому что она отличается от того, что было раньше, также у нас есть эта стременная штука, зашитая в формулу, которая зависит от w , параметров, по которым мы оптимизируем.

ФУНКЦИИ ОШИБКИ

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

Такая функция потерь была раньше, и мы разобрались, как брать от нее производную.

Переть посмотрим, как брать производную от этой штуки, потому что она отличается от того, что было раньше, также у нас есть эта стременная штука, зашитая в формулу, которая зависит от w , параметров, по которым мы оптимизируем.

ФУНКЦИИ ОШИБКИ

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

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

Такая функция потерь была раньше, и мы разобрались, как брать от нее производную.

Переть посмотрим, как брать производную от этой штуки, потому что она отличается от того, что было раньше, также у нас есть эта стременная штука, зашитая в формулу, которая зависит от w , параметров, по которым мы оптимизируем.

ФУНКЦИИ ОШИБКИ

раньше

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

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

Такая функция потерь была раньше, и мы разобрались, как брать от нее производную.

Переть посмотрим, как брать производную от этой штуки, потому что она отличается от того, что было раньше, также у нас есть эта стременная штука, зашитая в формулу, которая зависит от w , параметров, по которым мы оптимизируем.

ФУНКЦИИ ОШИБКИ

раньше

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

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

теперь

Такая функция потерь была раньше, и мы разобрались, как брать от нее производную.

Переть посмотрим, как брать производную от этой штуки, потому что она отличается от того, что было раньше, также у нас есть эта стременная штука, зашитая в формулу, которая зависит от w , параметров, по которым мы оптимизируем.

ФУНКЦИИ ОШИБКИ

раньше

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

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

теперь

тут w

Такая функция потерь была раньше, и мы разобрались, как брать от нее производную.

Переть посмотрим, как брать производную от этой штуки, потому что она отличается от того, что было раньше, также у нас есть эта стременная штука, зашитая в формулу, которая зависит от w , параметров, по которым мы оптимизируем.

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 дэдлайн: 17 ноября
 - После него будет даваться половина баллов
- Жесткий дедлайн: 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