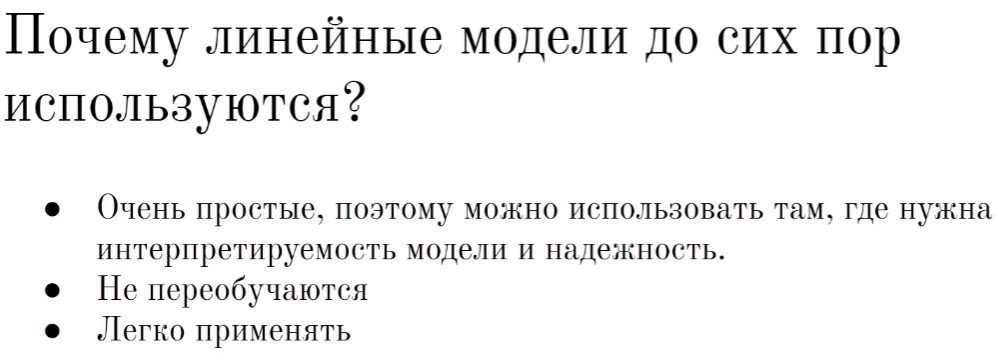
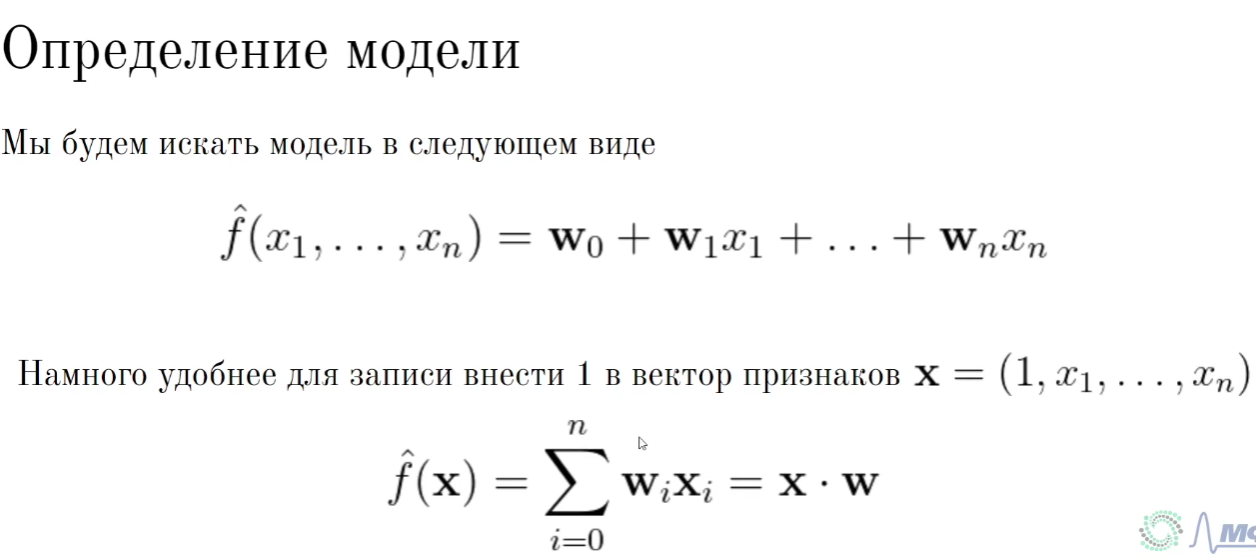
Линейные модели.

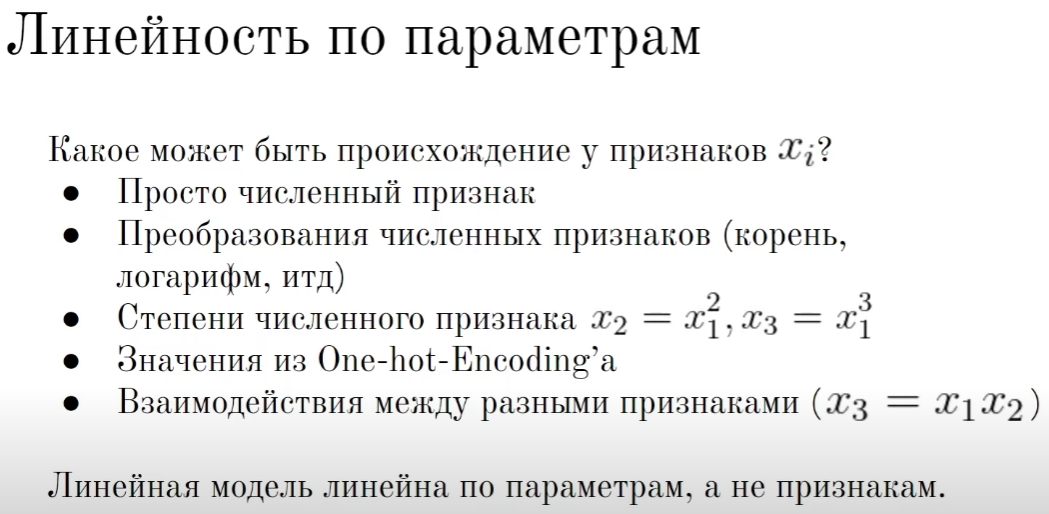


Линейные модели – универсальный способ решения задач машинного обучения. Хорошо работают там, где не очень много данных.

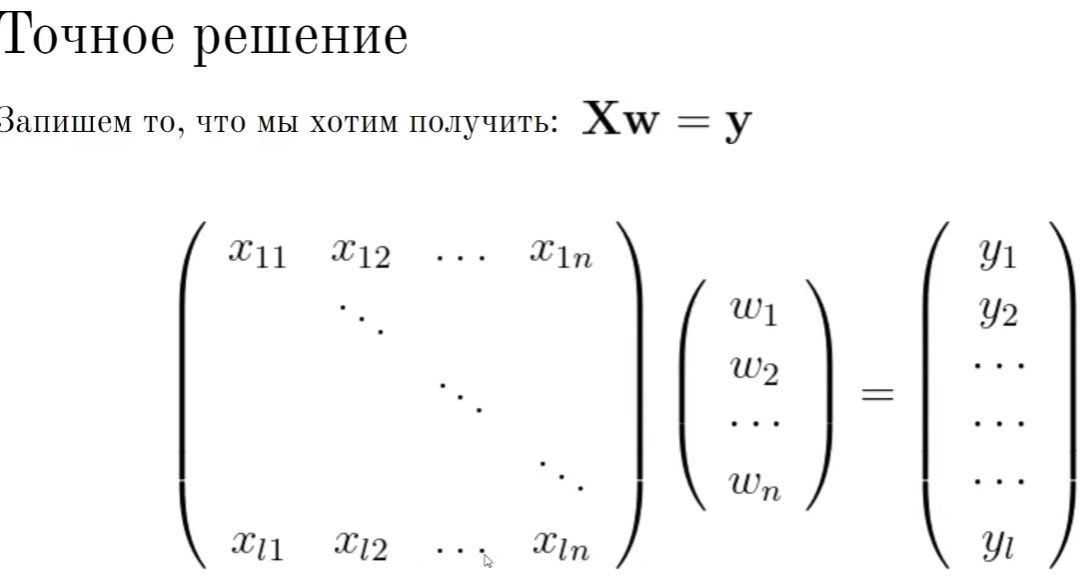


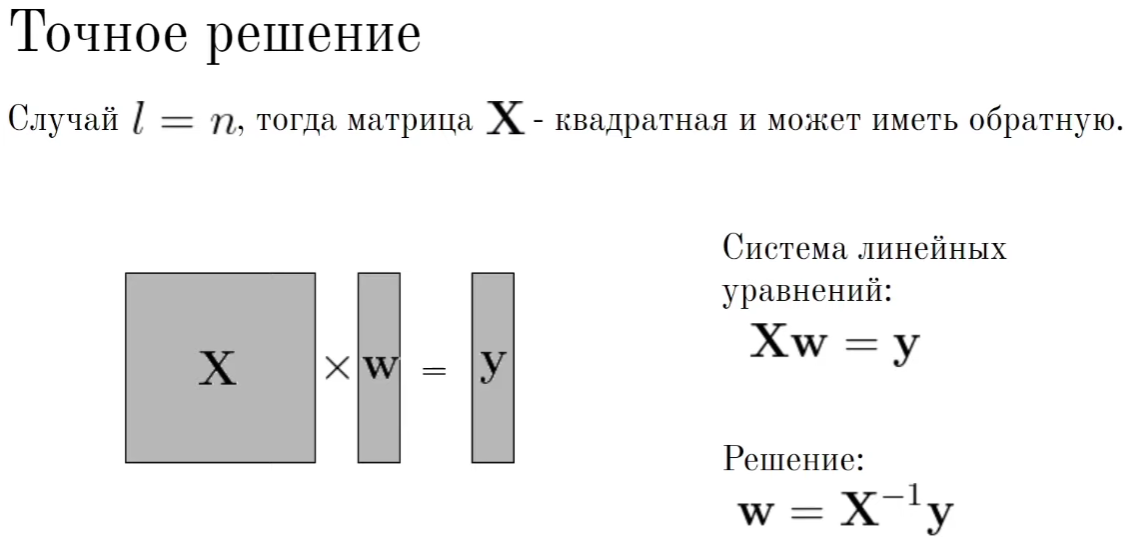
Для линейной регрессии используют функцию потерь квадрат ошибки, это самая часто встречающаяся функция.





Иногда в датасете бывают такие закономерности между признаками, которые линейная регрессия не может заметить. Чтобы она учитывала эти закономерности, мы можем в качестве признаков еще передавать степени каких то признаков и результаты взаимодействия между отдельным признаками (3 и 5 пункты из картинки выше).



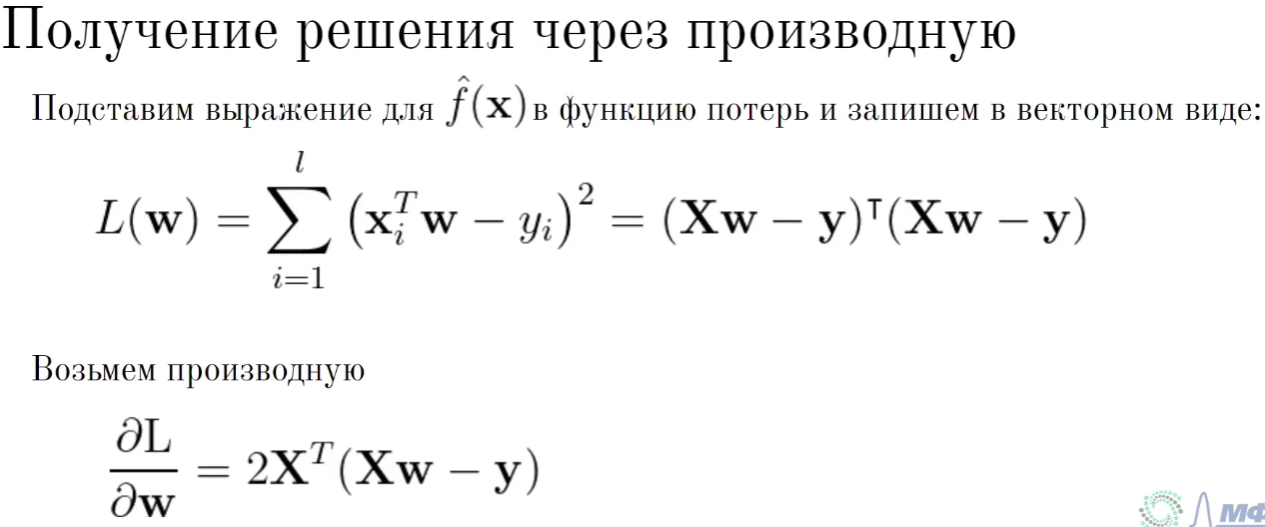


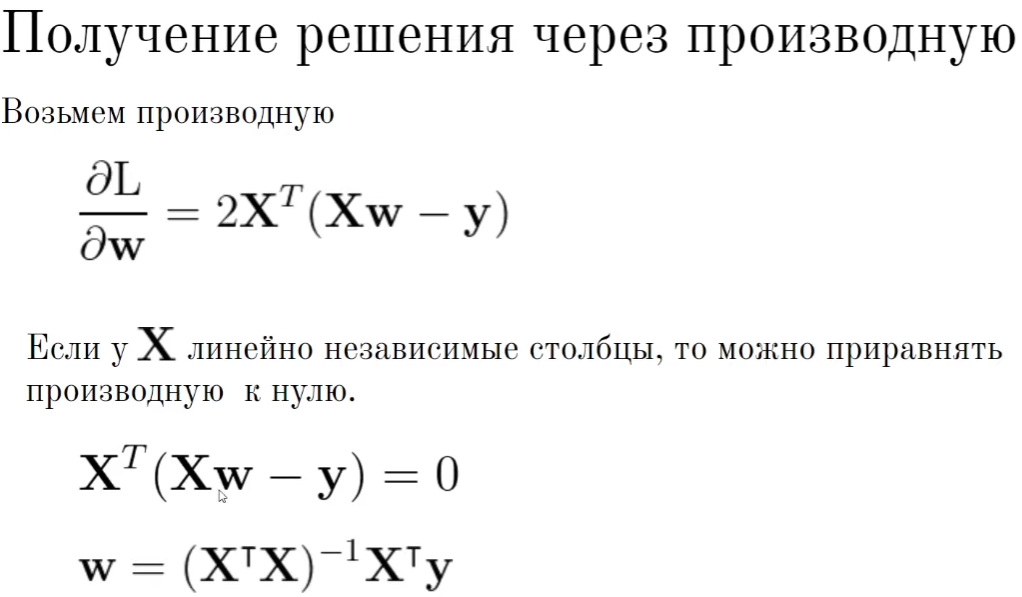
Но такой случай очень плохой, потому что наблюдений очень мало. Модель получится плохо обученной.



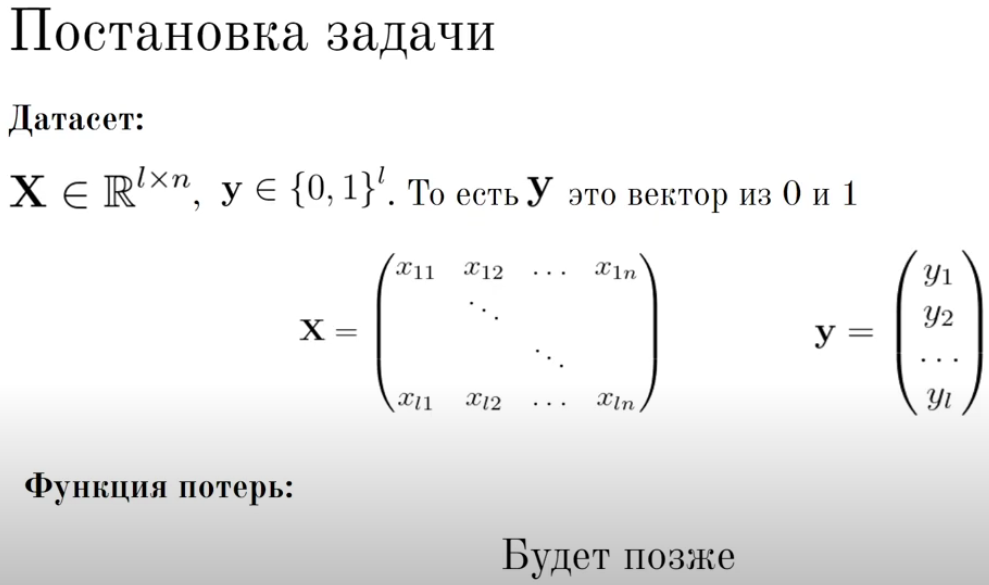
Если количество уравнений больше чем переменных, то уравнение не имеет решений. Поэтому мы находим матрицу весов, домножив обе части уравнений на псевдообратную матрицу X слева. Псевдообратная матрица устроена так, что вектор весов, который мы в итоге получаем, будет давать наименьший квадрат ошибки.

Если есть линейная зависимость между колонками, то ее нужно убрать, потому что иначе решение не получится.

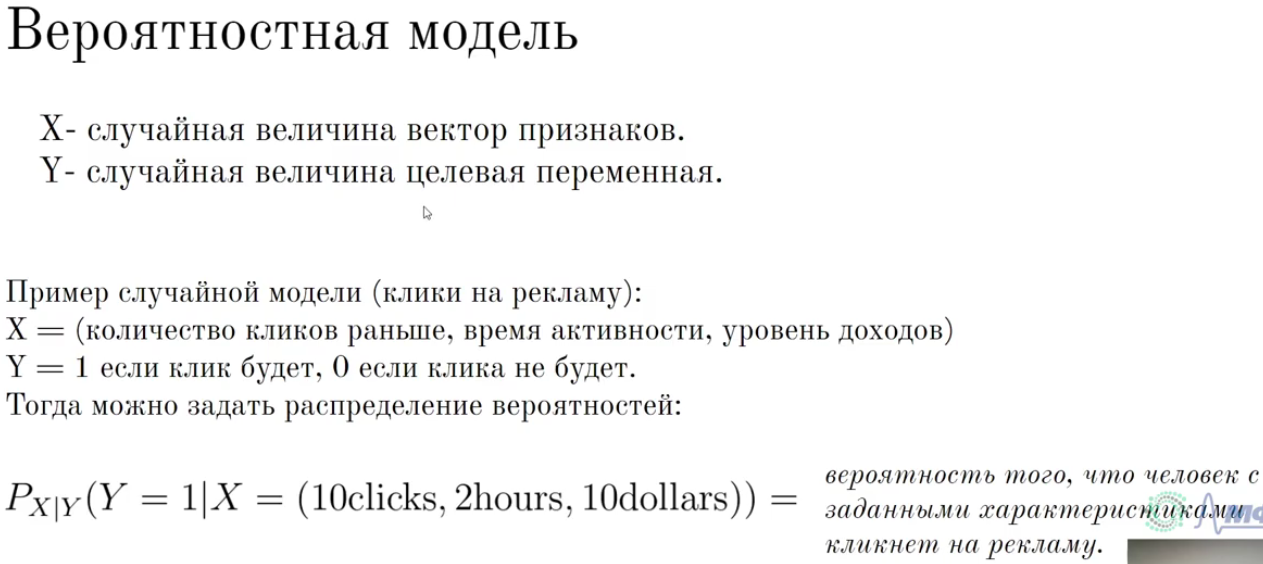




Задача классификации.



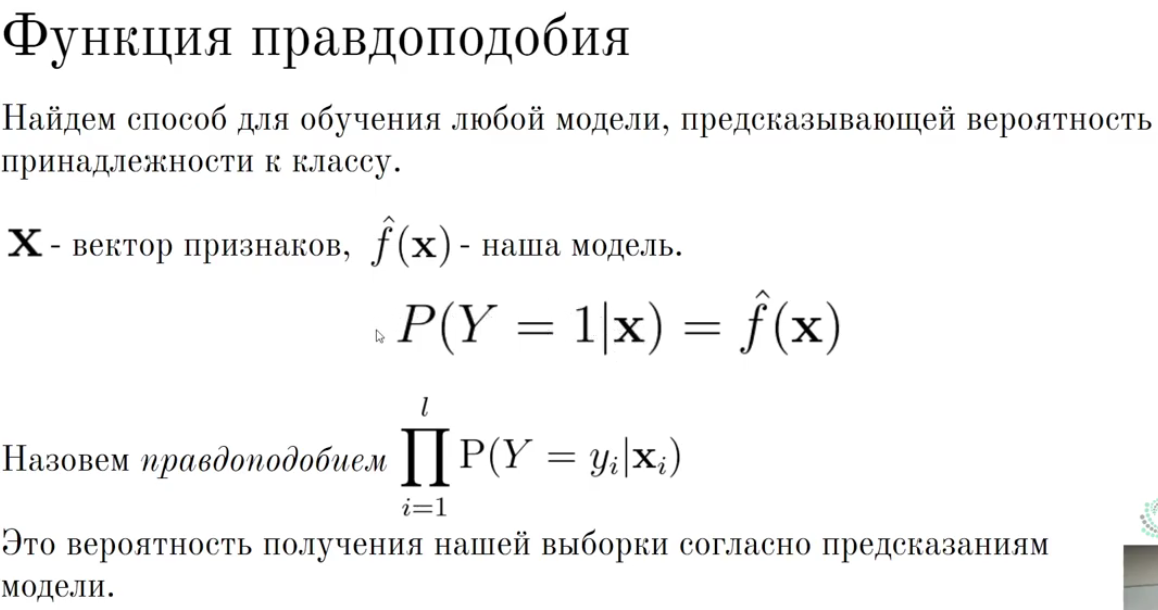
Классификация, в которой нужно определить произойдет ли данное событие, или нет, называется бинарной классификацией.



Вероятностная модель должна предсказывать вероятность.



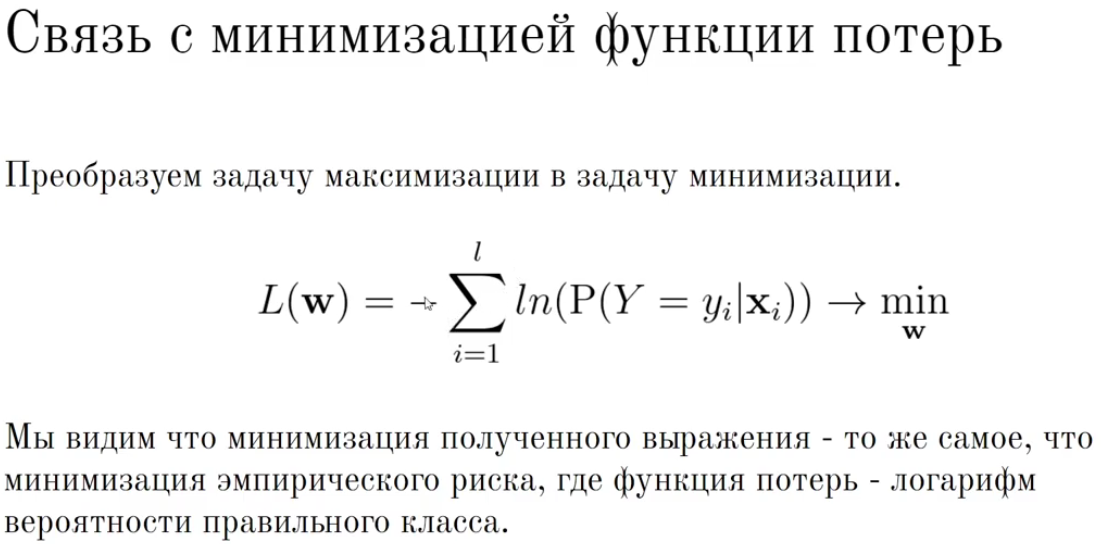
Эта запись означает, что наша функция P от X и Y возвращает вероятность того, что Y=1 при переданных признаках X.



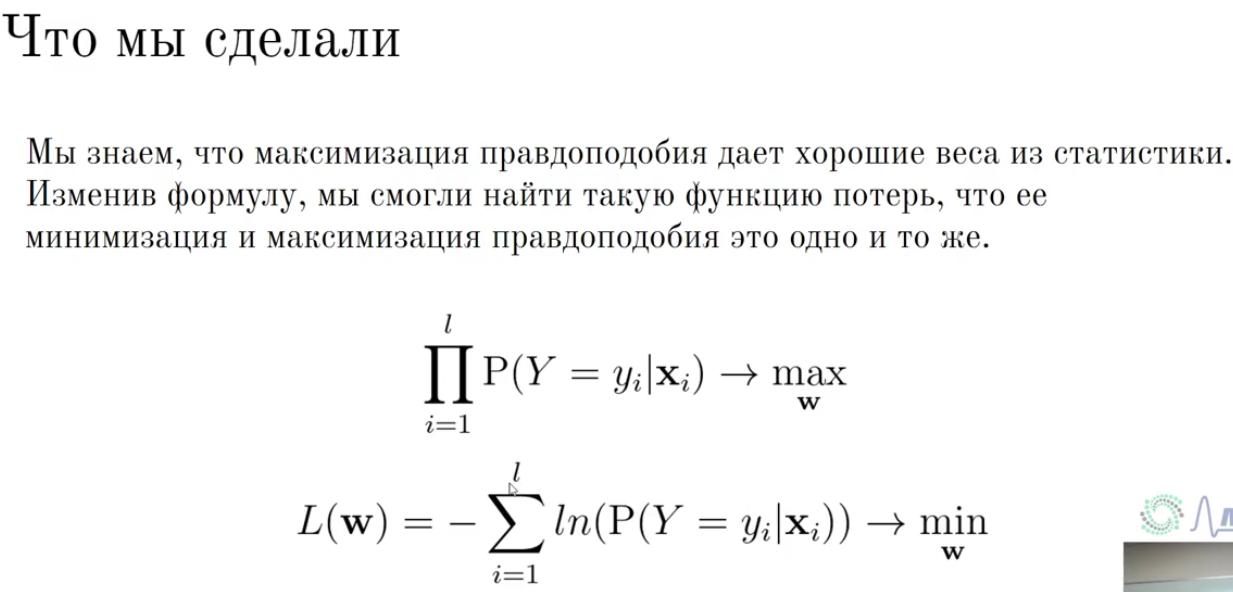
Функция правдоподобия – это произведение всех вероятностей для всех объектов выборки.



Вот это правдоподобие нам нужно максимизировать, потому что чем оно больше, тем ближе к правдоподобным результатам выдает результаты наша модель. Мы берем логарифм от функции произведения, потому что поведение функции не меняется, т.е. все своейства функции остаются, но теперь подсчет становится гораздо проще, потому что логарифм произведения можно разбить на сумму логарифмов.



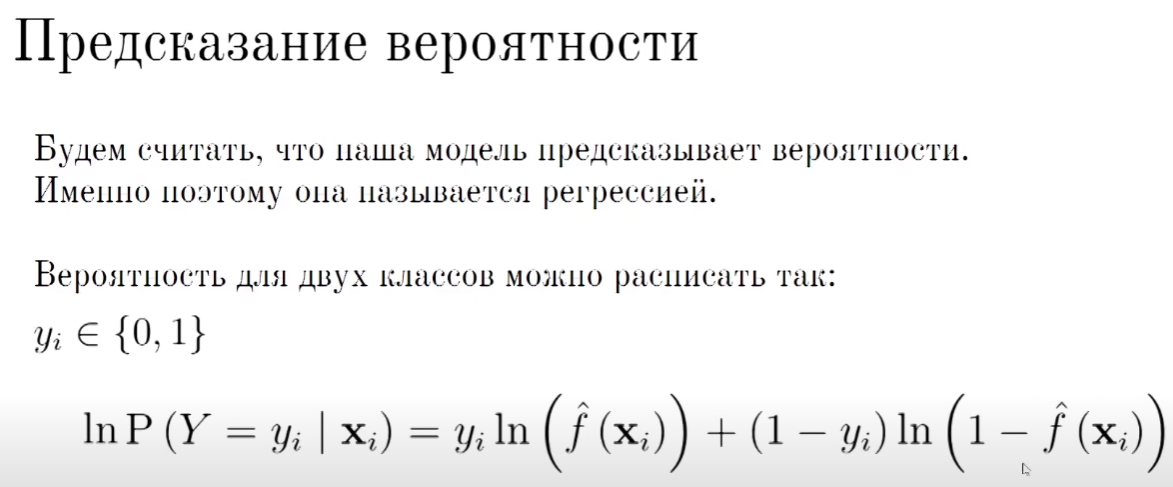
Здесь формула выглядит в простом виде, мы просто заменили функцию произведения на сумму логарифмов, посмотри просто предыдущую картинку, там без логарифма формула есть.

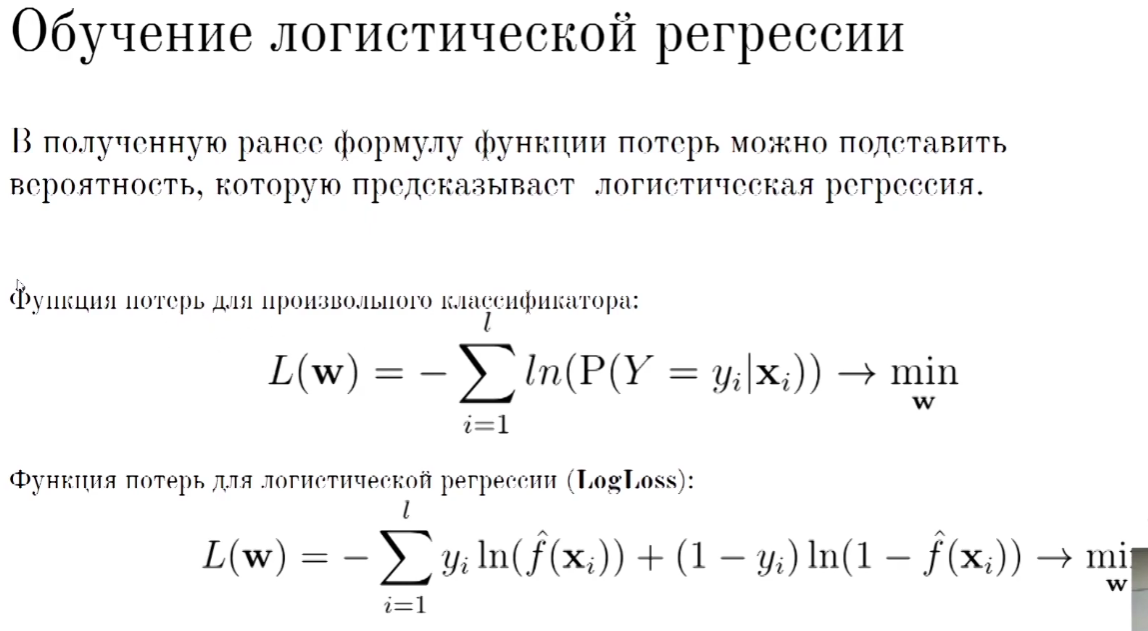


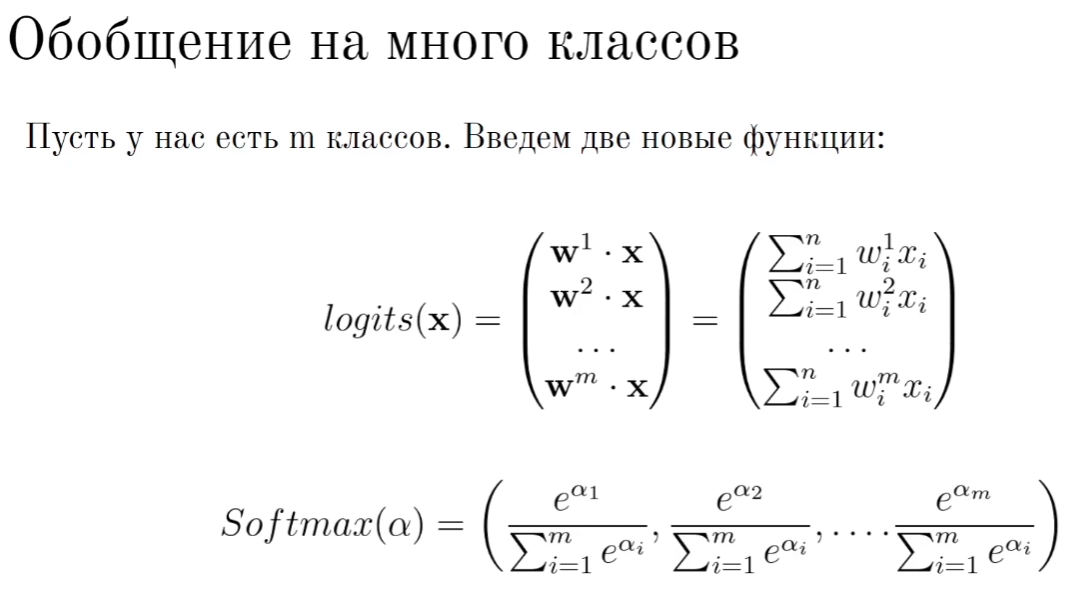
**Логистическая регрессия.**

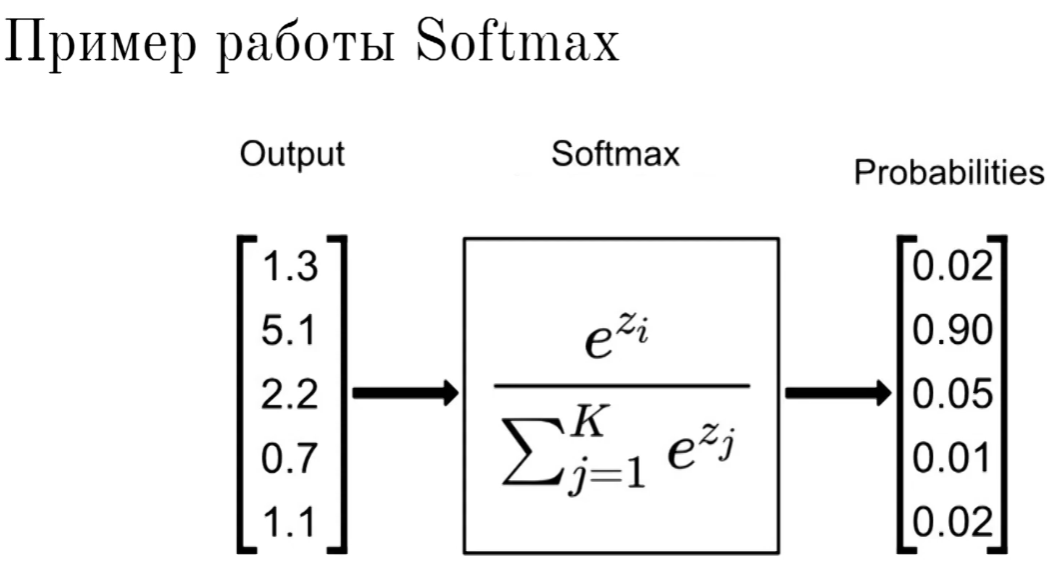


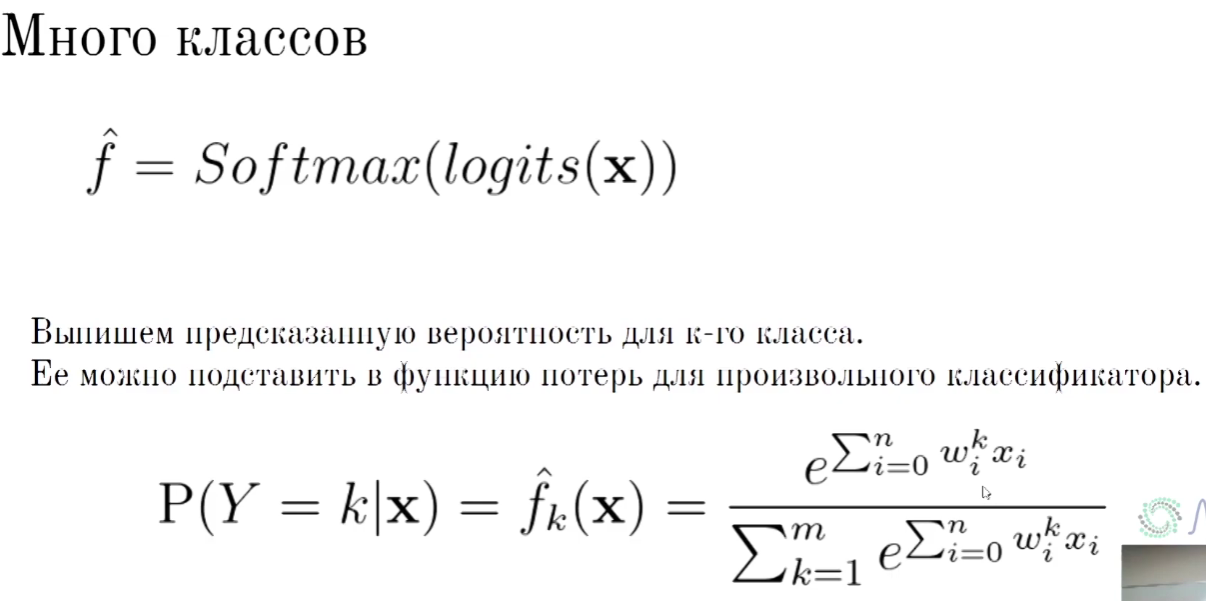
Функция сигмоиды – это функция, которая обрабатывает аргумент и возвращает значение от 0 до 1. Фактически, функция сигмоиды возвращает вероятность. Она преобразовывает ответ из промежутка (-inf, +inf) в промежуток (0,1).



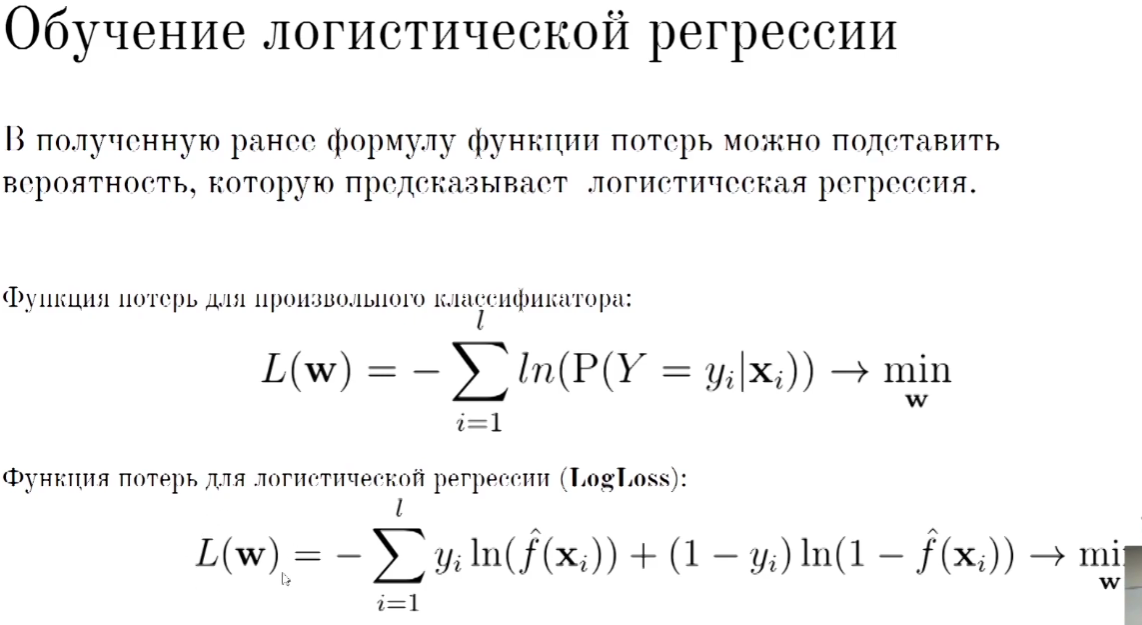




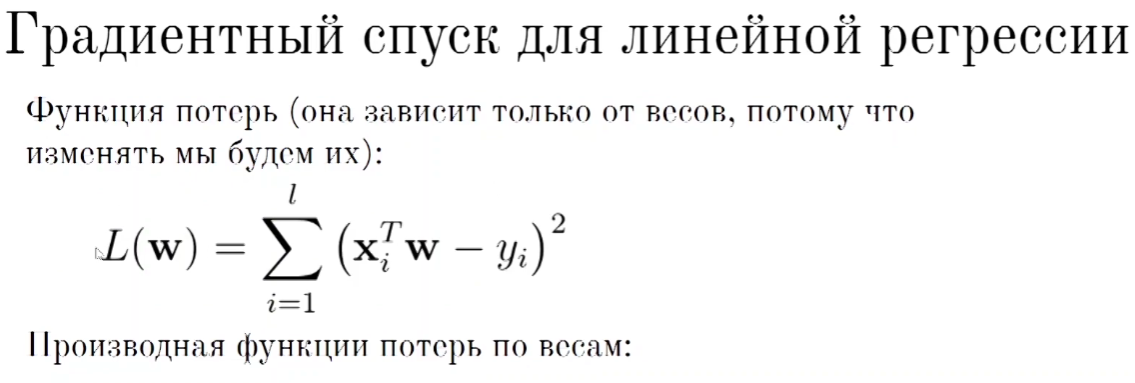


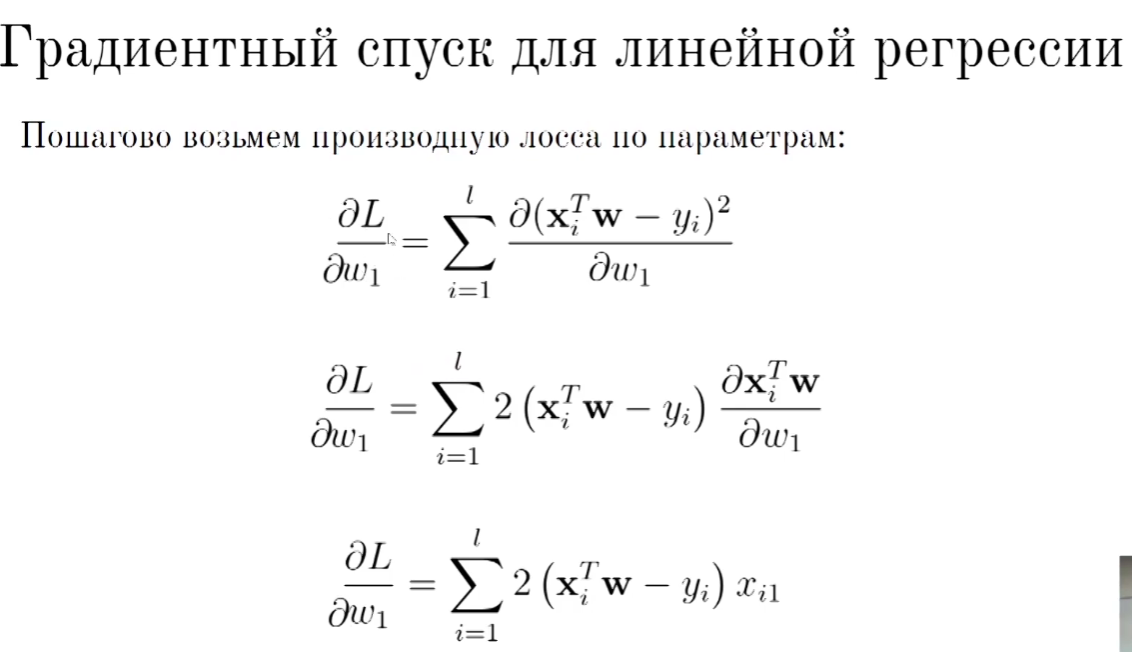


**Градиентный спуск в линейных моделях.**

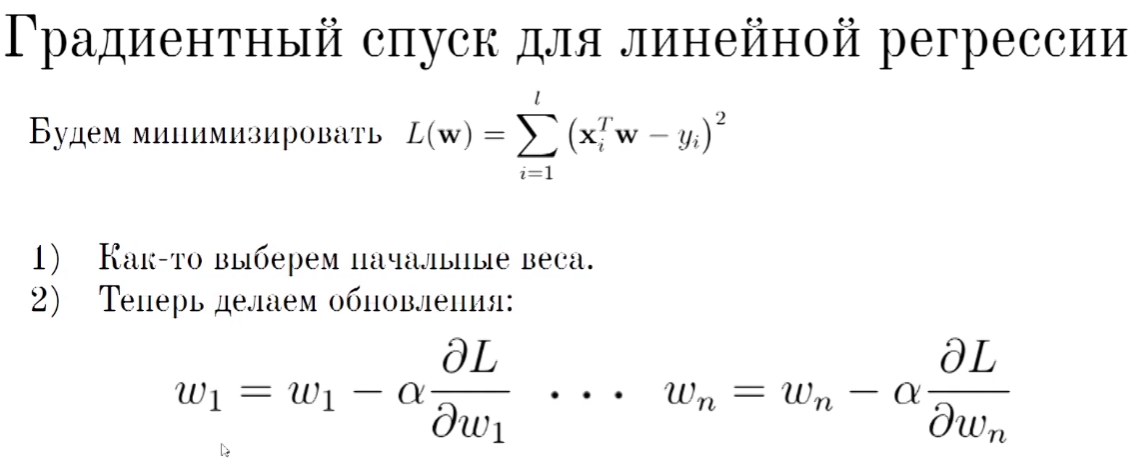


**Градиентный спуск в линейных алгоритмах.**

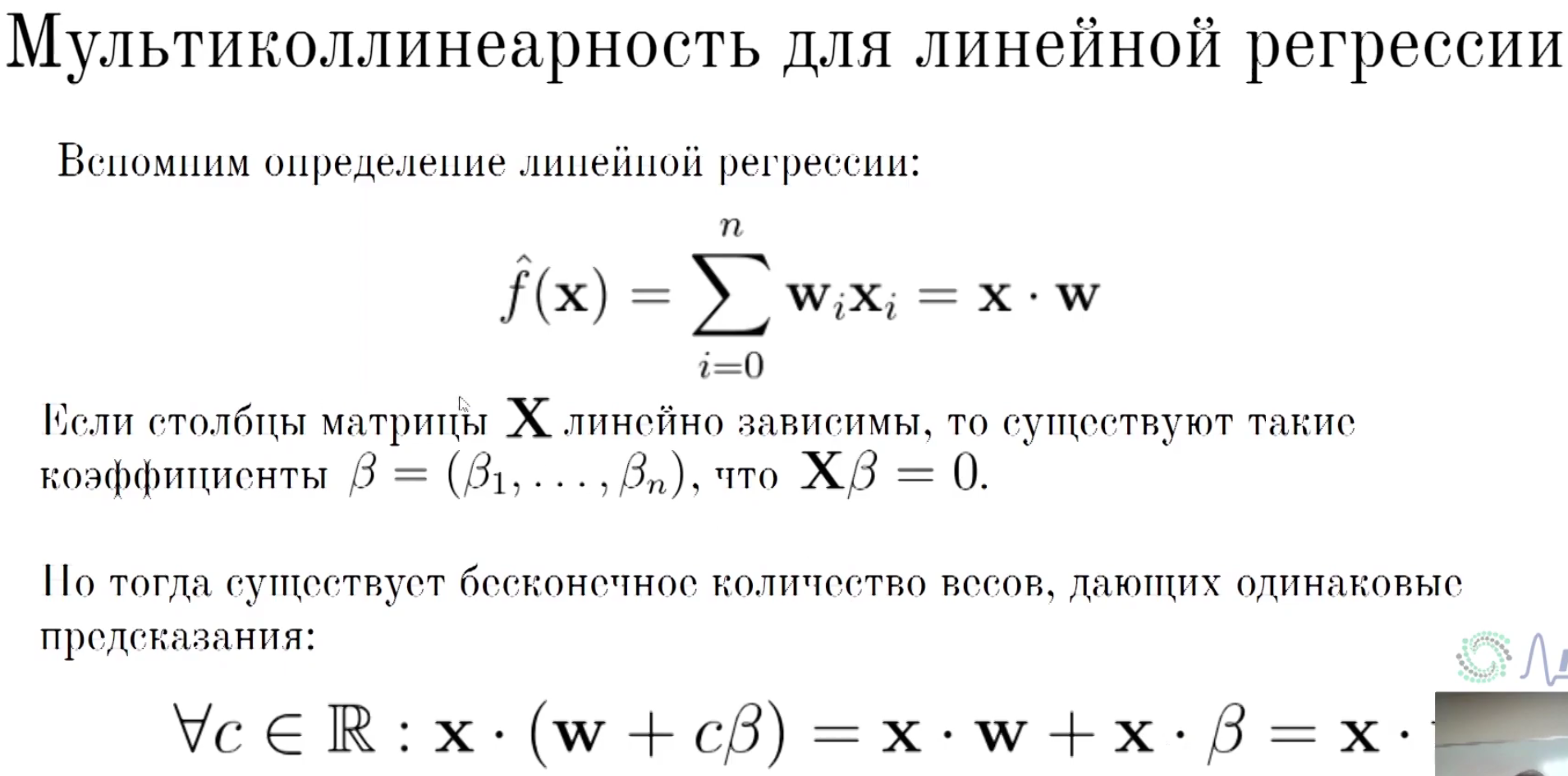


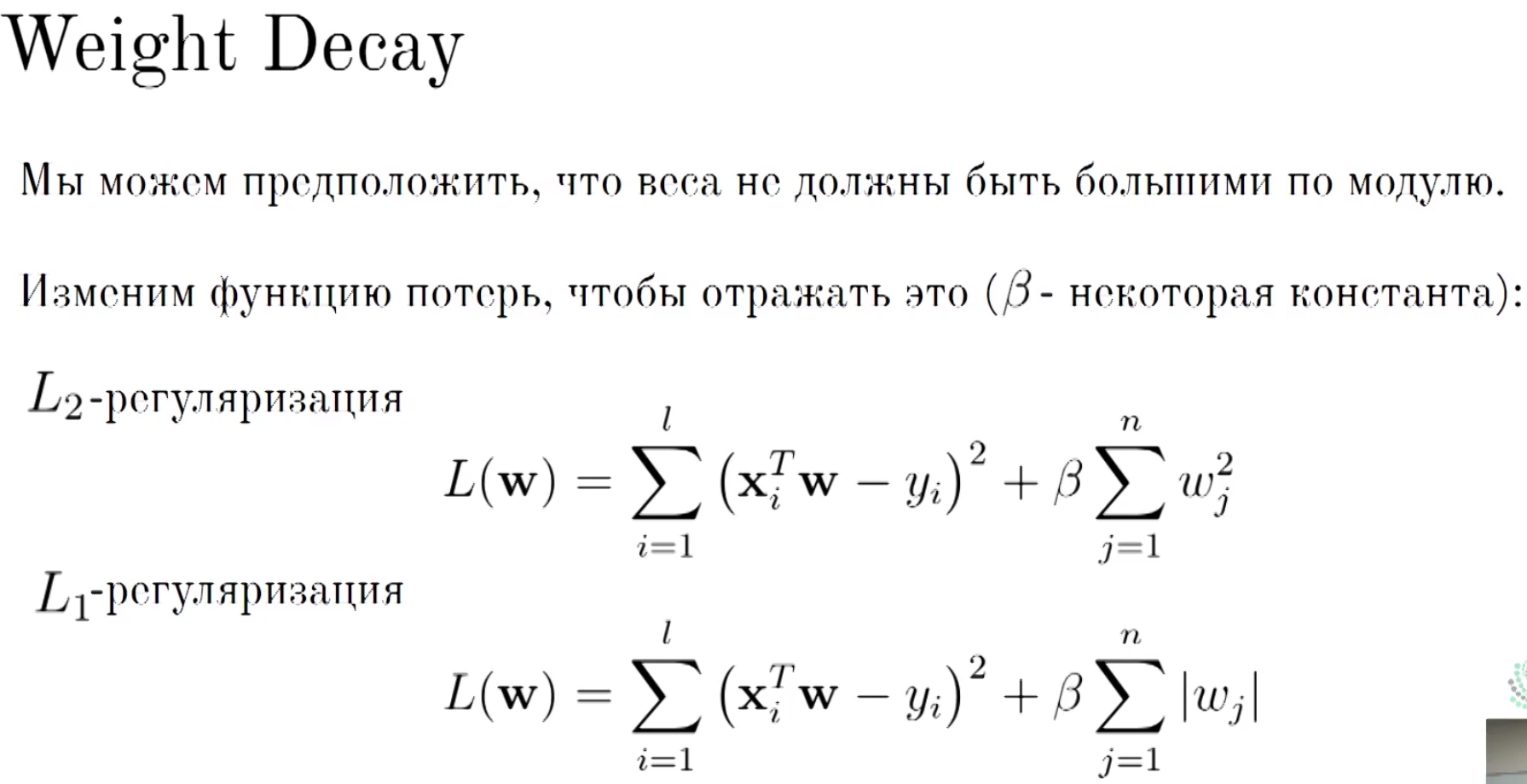


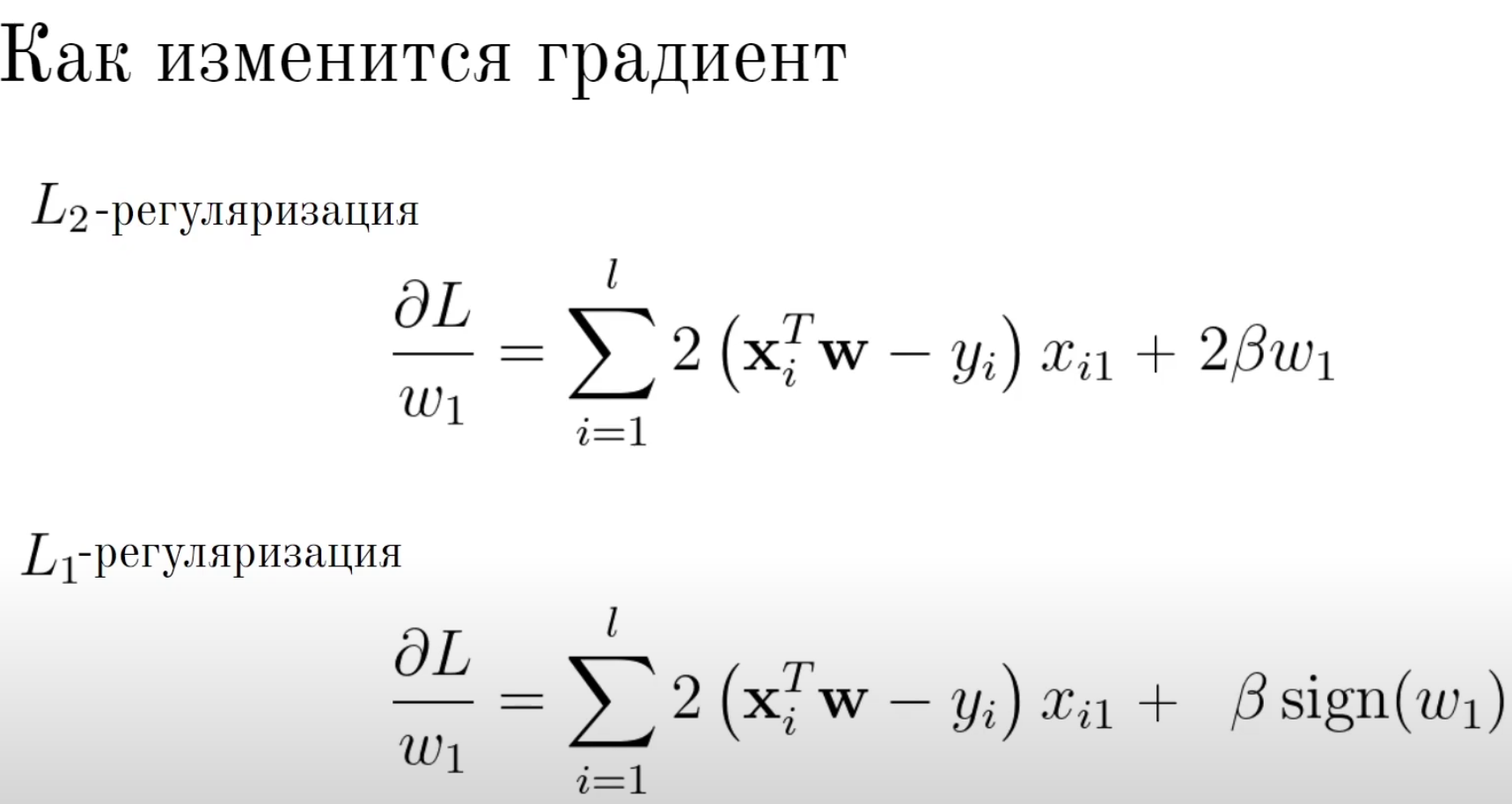
Как обучается модель: мы подбираем определенные веса.



Регуляризация – это такой процесс. Может быть такое, что линейная модель, вследствие линейной зависимости ее строк, переобучилась. И веса, которые мы в получили в лоссе оказались какими то огромными. Регуляризация позволяет добавить в наш алгоритм знание, что веса по модулю не должны быть большими.

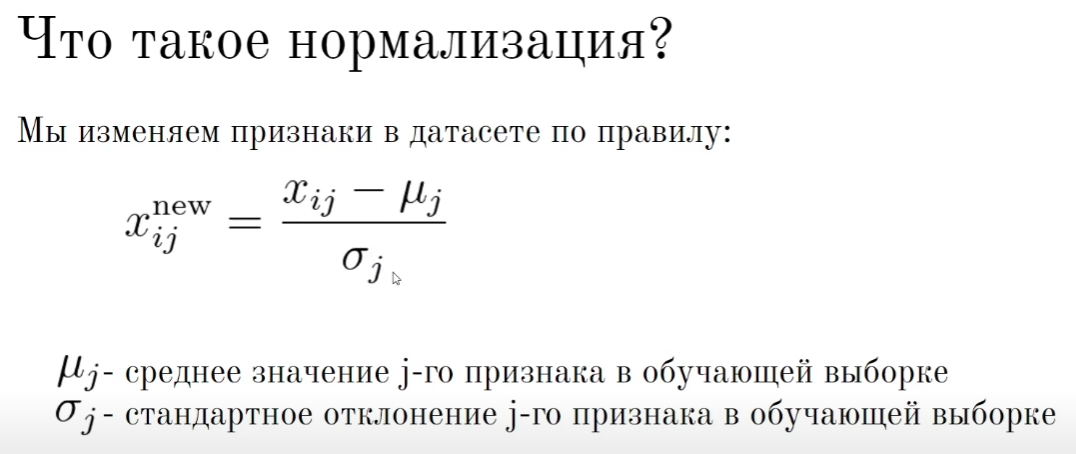




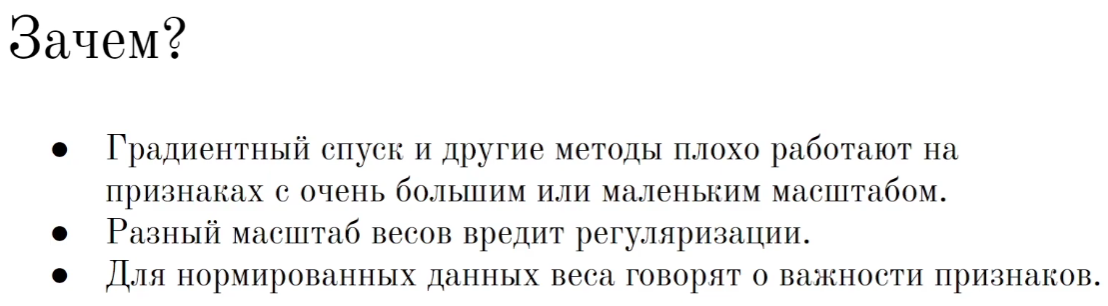


**Нормализация признаков.**

Нужно для придания одинакового масштаба всему датасету.



Стандартное отклонение – это среднее значение от разности среднего значения выборки и различных элементов выборки.



При работе с любой линейной моделью и с любой нейронной сетью нужно всегда подавать на вход отнормированные данные, иначе все может поломаться.

**Эмпирический риск** (Empirical Risk) — это средняя величина ошибки алгоритма на обучающей выборке.