Введение в
Машинное обучение

Александр Безносиков

ИСП РАН

13 февраля 2025

Обучение с учителем (Supervised learning)

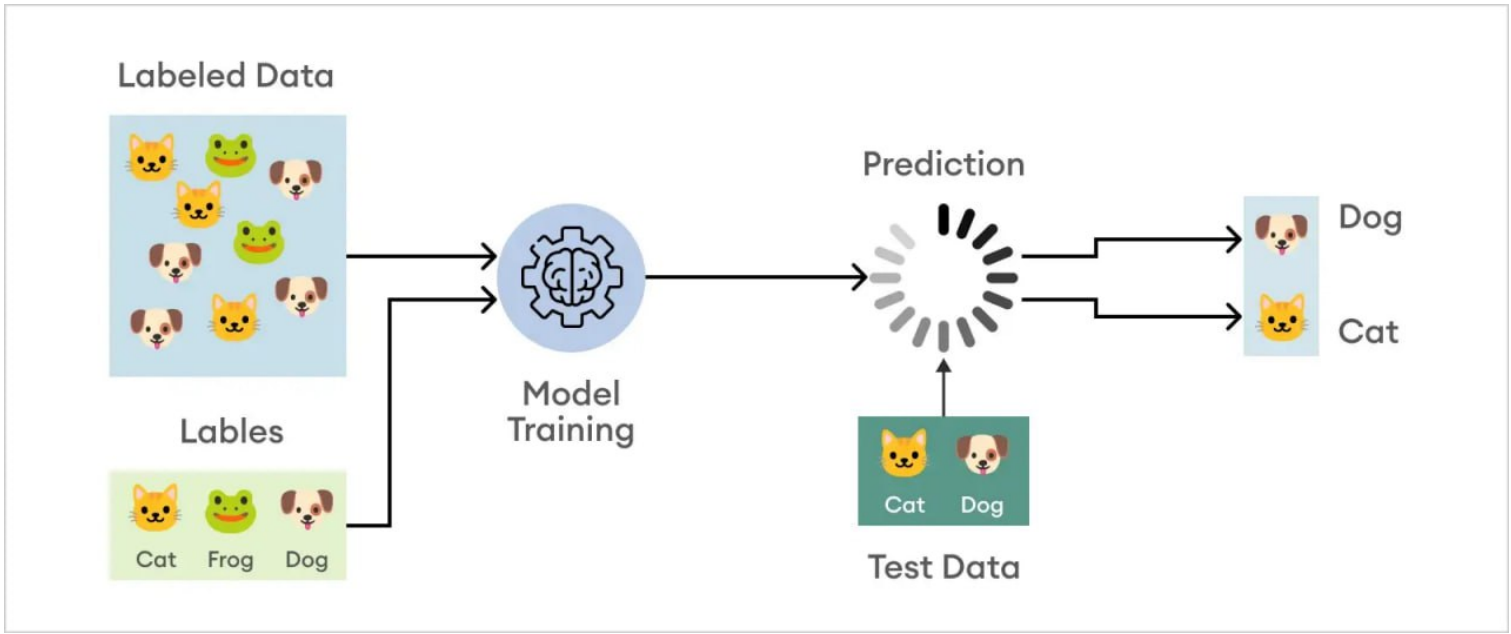
На вход подается выборка X, состоящая из пар (x_i, y_i):

$$X = \{(x_1, y_1), \ldots, (x_n, y_n)\}.$$

- $y_i \in \mathcal{Y}$ – значение целевой функции (переменная/target/dependent variavle), $\mathcal{Y}$ – пространство меток (значений целевого признака);
- $x_i \in \mathcal{X}$ – объект (наблюдение/sample/instance), $\mathcal{X}$ – пространство объектов (входов);
- Специфика задач обучения состоит в том, что нам известны значения меток y_i.

Обучение с учителем (Supervised learning)

Основной целью является восстановление зависимости $g : \mathcal{X} \to \mathcal{Y}$, чтобы уметь восстанавливать метки на новых объектах x.



Типы задач обучения с учителем

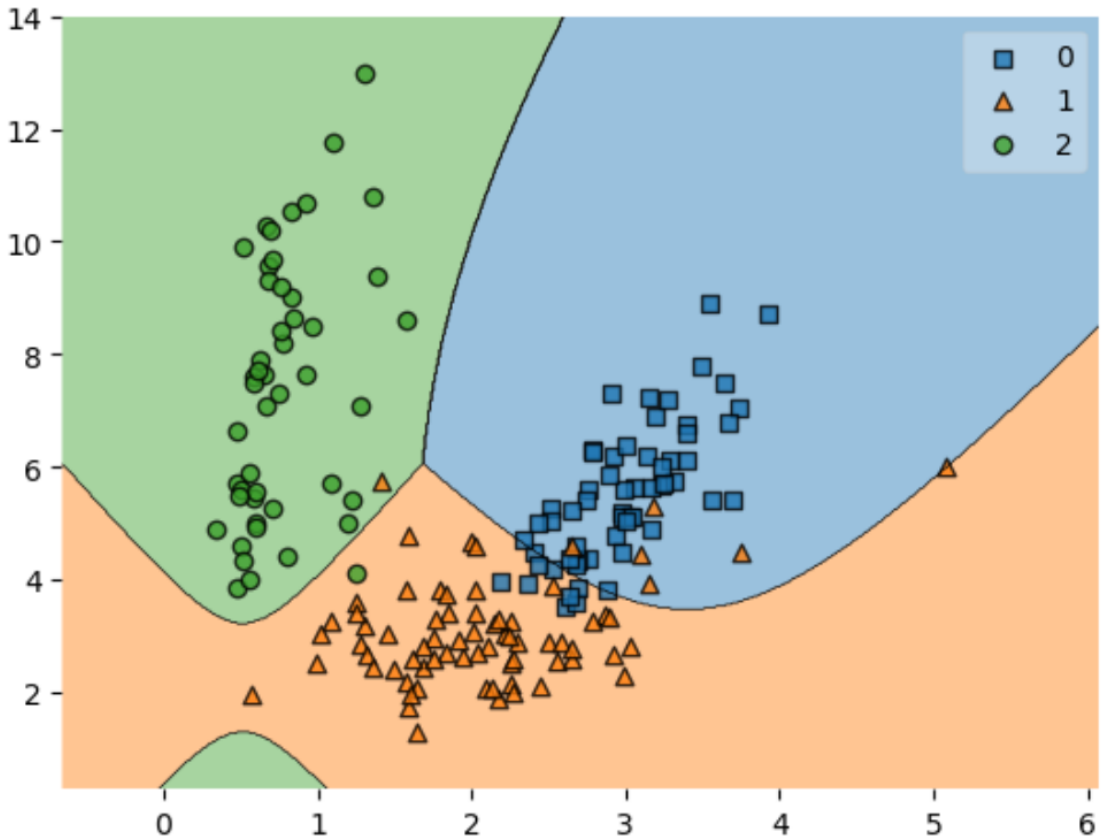
Классификация (Classification)

Множество $\mathcal{Y}$ дискретно и $|\mathcal{Y}| = k \ll \infty$.

1. Бинарная: $\mathcal{Y} = \{0, 1\}$ или $\mathcal{Y} = \{-1, +1\}$ (binary classification);
2. k *непересекающихся* классов: $\mathcal{Y} = \{0, 1, \ldots, k\}$ (multiclass classification);
3. k *пересекающихся* классов: $\mathcal{Y} = \{0, 1\}^k$ (multilabel classification).

Типы задач обучения с учителем

Классификация (Classification)

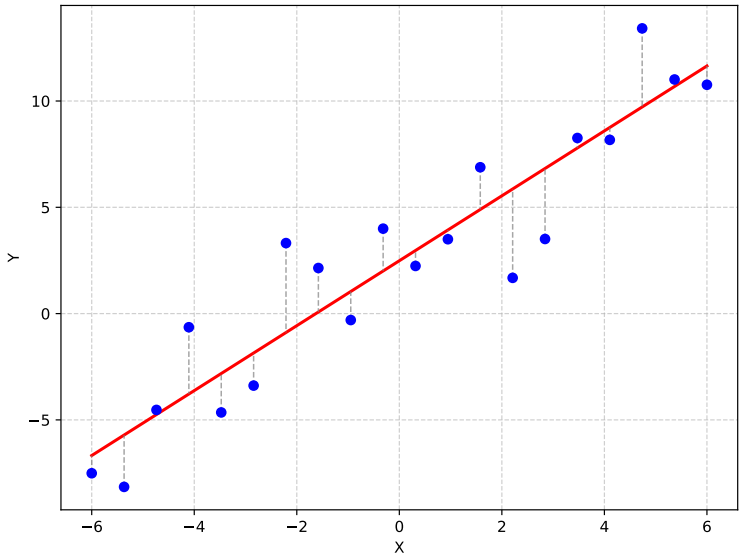


Типы задач обучения с учителем

Регрессия (regression)

Множество $\mathcal{Y}$ непрерывно:

- Одномерная линейная регрессия: $\mathcal{Y} = \mathbb{R}$ (linear regression).

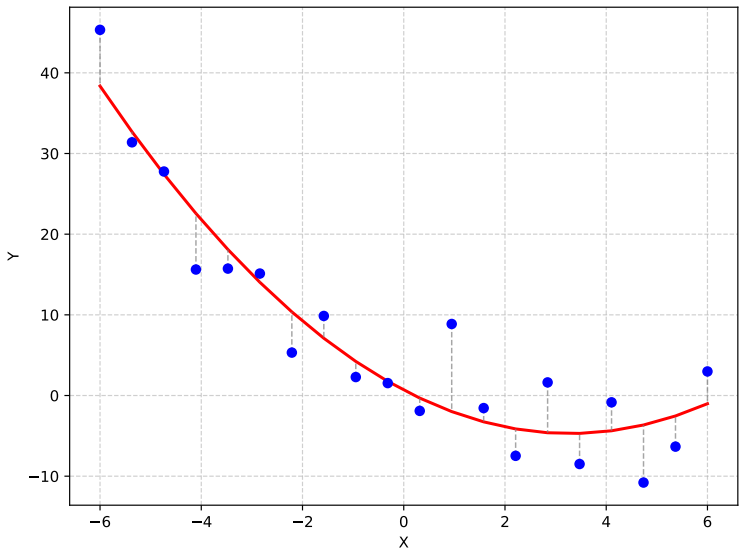


Типы задач обучения с учителем

Регрессия (regression)

Множество $\mathcal{Y}$ непрерывно:

- Одномерная полиномиальная регрессия: $\mathcal{Y} = \mathbb{R}$ (polynomial).

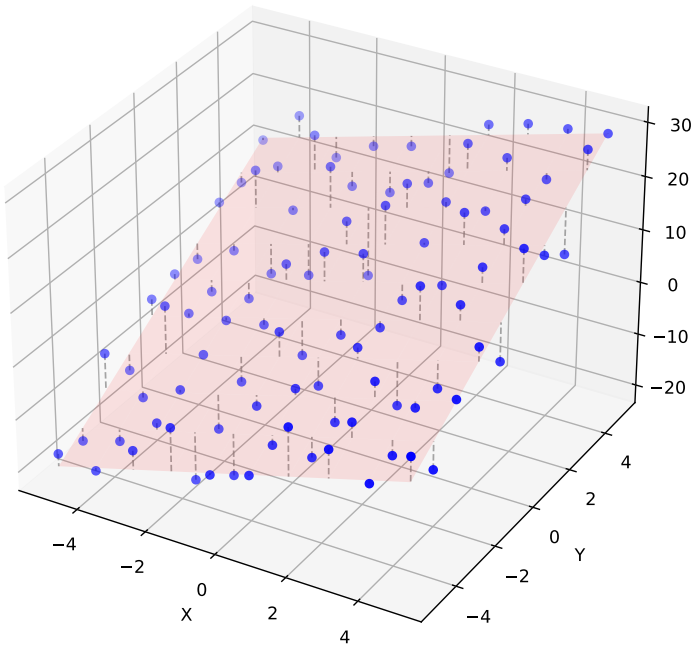


Типы задач обучения с учителем

Регрессия (regression)

Множество $\mathcal{Y}$ непрерывно:

- Многомерная регрессия: $\mathcal{Y} = \mathbb{R}^n$ (multi-dimensional regression).



Пространство объектов

Объекты могут быть почти произвольными:

- тексты;
- временные ряды/последовательности;
- изображения (как 2D, так и 3D/4D);
- вектора/графы;
- ...

Признаковое описание

В признаковом описании, i-ый объект имеет следующий вид:

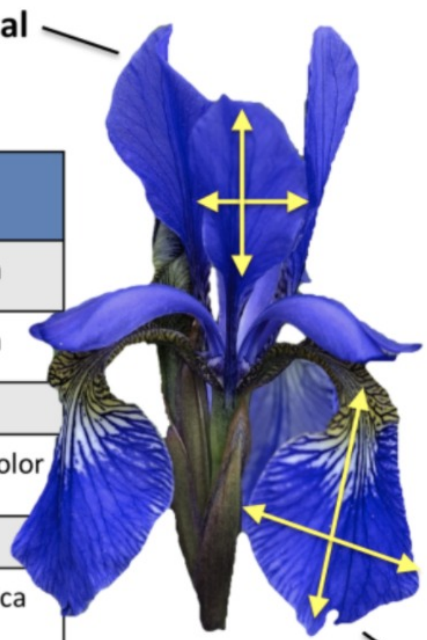
$$x_i = (x_{i1}, \ldots, x_{id}) \in \mathbb{R}^d,$$

где x_{ij} – j-ый признак i-го объекта.

Матрица «объект-признак» (data matrix)

Samples
(instances, observations)

	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica



Features
(attributes, measurements, dimensions)

Class labels
(targets)

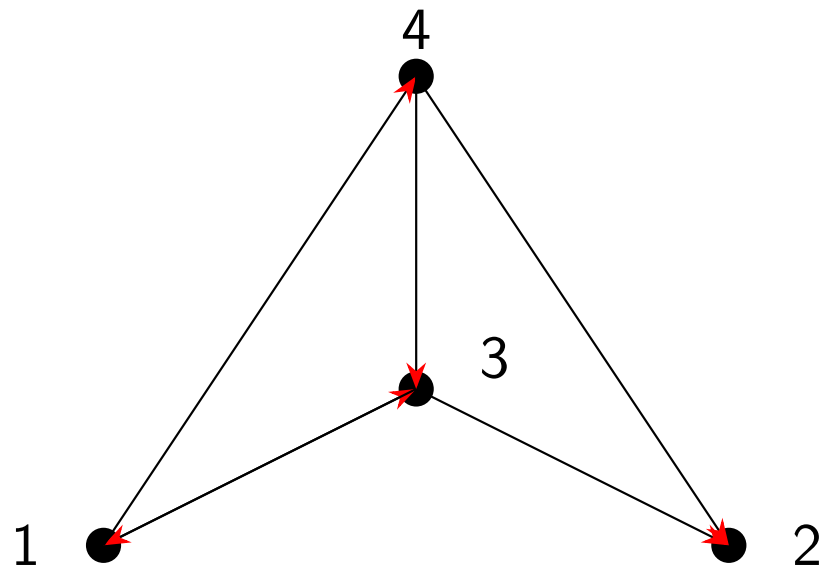
Генерация признаков

Необходимость признаков

Чем *больше* признаков у объекта – тем *проще* ML подход к ее решению.

Но объекты могут не быть заданы в признаковом пространстве, либо заданы неинформативно. Например, графы.

- Степень вершины;
- Ориентация ребра;
- Достижимость;
- Сток/исток.



Генерация признаков (примеры)

- **Классификация спама**: $\mathcal{X}$ – письма, $\mathcal{Y}$ – бинарная велична (спам/не спам), признаки – длина письма, число вхождений слова, отправитель, . . .;
- **Медицинская диагностика**: $\mathcal{X}$ – пациенты, $\mathcal{Y}$ – диагнозы, признаки – результаты анализов, возраст, пол, . . .

Математическая постановка задачи обучения: модель и веса

- Природа может быть слишком сложной, чтобы ее полноценно описать, ограничим поиск отображения из $g(\cdot) : \mathcal{X} \to \mathcal{Y}$.

Математическая постановка задачи обучения: модель и веса

- Природа может быть слишком сложной, чтобы ее полноценно описать, ограничим поиск отображения из $g(\cdot) : \mathcal{X} \to \mathcal{Y}$. **Вопрос:** каким образом можно ограничить поиск g, при этом хочется удобства перебора по всем таким g?

Математическая постановка задачи обучения: модель и веса

- Природа может быть слишком сложной, чтобы ее полноценно описать, ограничим поиск отображения из $g(\cdot) : \mathcal{X} \to \mathcal{Y}$. **Вопрос:** каким образом можно ограничить поиск g, при этом хочется удобства перебора по всем таким g?
- Введем параметризацию $g(\cdot, w)$, зависящую от **вектора весов (parameters)** w. Примеры:
  1. $g(x, w) = w_0 + x_1 \cdot w_1 + x_2 \cdot w_2$.

Математическая постановка задачи обучения: модель и веса

- Природа может быть слишком сложной, чтобы ее полноценно описать, ограничим поиск отображения из $g(\cdot) : \mathcal{X} \to \mathcal{Y}$. **Вопрос:** каким образом можно ограничить поиск g, при этом хочется удобства перебора по всем таким g?
- Введем параметризацию $g(\cdot, w)$, зависящую от **вектора весов (parameters)** w. Примеры:
  1. $g(x, w) = w_0 + x_1 \cdot w_1 + x_2 \cdot w_2$.
  2. $g(x, w) = \begin{cases} 1, & \text{если} \quad x_1 + w_1 > 10, \\ 0, & \text{иначе}, \end{cases}$

Математическая постановка задачи обучения: модель и веса

- Природа может быть слишком сложной, чтобы ее полноценно описать, ограничим поиск отображения из $g(\cdot) : \mathcal{X} \to \mathcal{Y}$. **Вопрос:** каким образом можно ограничить поиск g, при этом хочется удобства перебора по всем таким g?
- Введем параметризацию $g(\cdot, w)$, зависящую от **вектора весов (parameters)** w. Примеры:
  1. $g(x, w) = w_0 + x_1 \cdot w_1 + x_2 \cdot w_2$.
  2. $g(x, w) = \begin{cases} 1, & \text{если} \quad x_1 + w_1 > 10, \\ 0, & \text{иначе}, \end{cases}$
  3. часто $g(\cdot, w)$ – это композиция более атомарных функций: $g(\cdot, w) = g_m(\dots g_2(g_1(\cdot, w_1), w_2), \dots w_m)$.
- $g(\cdot, w)$ мы будем называть **моделью машинного обучения**.

Математическая постановка задачи обучения: функция потерь

- Глобально мы бы хотели, чтобы для любой пары (объект x, ответ y) наша модель с весами w отвечала бы следующему свойству: $g(x, w) = y$ (верно бы угадывала ответ).
- Модель с произвольными весами w очевидно может не дать хоть немного правильные предсказания природы. Чтобы "научить" модель (подобрать веса/параметры) нужно измерить степень непохожести ответов: модели и реального.
- Введем функцию потерь (loss) $\ell(\cdot, \cdot)$, зависящую от двух аргументов. Примеры
  1. $\ell(y_1, y_2) = (y_1 - y_2)^2$.
  2. $\ell(y_1, y_2) = \begin{cases} 1, & \text{если} \quad y_1 \neq y_2, \\ 0, & \text{иначе}, \end{cases}$
- Нас будет интересовать: $\ell(g(x, w), y)$.

Математическая постановка задачи обучения: формулировка

- **Вопрос:** Формально откуда берутся данные? Они приходят из некоторой "природы" (случайного распределения) $\mathcal{D}$.

Цель supervised обучения

Формальная цель supervised машинного обучения – найти природу $\mathcal{D}$ в точности или приблизить ее с помощью g.

Математическая постановка задачи обучения: формулировка

- **Вопрос:** Формально откуда берутся данные? Они приходят из некоторой "природы" (случайного распределения) $\mathcal{D}$.

Цель supervised обучения

Формальная цель supervised машинного обучения – найти природу $\mathcal{D}$ в точности или приблизить ее с помощью g.

- Тогда задачу машинного обучения можно сформулировать следующим образом:

Математическая постановка задачи supervised обучения

$$\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\ell(g(x,w),y)\right]$$

- Получается, что мы хотим минимизировать потери модели в среднем по всей природе.

Математическая постановка задачи обучения: формулировка

- **Вопрос:** В чем проблема решения задачи оптимизации $\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} [\ell(g(x, w), y)]$?

Математическая постановка задачи обучения: формулировка

- **Вопрос:** В чем проблема решения задачи оптимизации $\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} [\ell(g(x, w), y)]$? $\mathcal{D}$ неизвестна, поэтому интеграл (математичсеское ожидание) никак не посчитать.

Математическая постановка задачи обучения: формулировка

- **Вопрос:** В чем проблема решения задачи оптимизации $\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} [\ell(g(x, w), y)]$? $\mathcal{D}$ неизвестна, поэтому интеграл (математичсеское ожидание) никак не посчитать. **Вопрос:** Как быть? Что обычно есть вместо $\mathcal{D}$?
- Обычно есть некоторая выборка (data sample):

$$\{x_i, y_i\}_{i=1}^n \sim \mathcal{D}.$$

Математическая постановка задачи обучения: формулировка

- **Вопрос:** В чем проблема решения задачи оптимизации $\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} [\ell(g(x, w), y)]$? $\mathcal{D}$ неизвестна, поэтому интеграл (математичсеское ожидание) никак не посчитать. **Вопрос:** Как быть? Что обычно есть вместо $\mathcal{D}$?
- Обычно есть некоторая выборка (data sample):
$$\{x_i, y_i\}_{i=1}^n \sim \mathcal{D}.$$
- Тогда интеграл с предыдущего слайда можно приблизить с помощью следующего выражения:

Минимизация эмпирического риска (ERM)

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(g(x_i, w), y_i)$$

Математическая постановка задачи обучения: формулировка

- **Вопрос:** В чем проблема решения задачи оптимизации $\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} [\ell(g(x, w), y)]$? $\mathcal{D}$ неизвестна, поэтому интеграл (математичсеское ожидание) никак не посчитать. **Вопрос:** Как быть? Что обычно есть вместо $\mathcal{D}$?
- Обычно есть некоторая выборка (data sample):
$$\{x_i, y_i\}_{i=1}^n \sim \mathcal{D}.$$
- Тогда интеграл с предыдущего слайда можно приблизить с помощью следующего выражения:

Минимизация эмпирического риска (ERM)

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(g(x_i, w), y_i)$$

Вопрос: Как называется такой способ приближения интеграла?

Математическая постановка задачи обучения: формулировка

- **Вопрос:** В чем проблема решения задачи оптимизации $\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}} [\ell(g(x, w), y)]$? $\mathcal{D}$ неизвестна, поэтому интеграл (математичсеское ожидание) никак не посчитать. **Вопрос:** Как быть? Что обычно есть вместо $\mathcal{D}$?
- Обычно есть некоторая выборка (data sample):
$$\{x_i, y_i\}_{i=1}^n \sim \mathcal{D}.$$
- Тогда интеграл с предыдущего слайда можно приблизить с помощью следующего выражения:

Минимизация эмпирического риска (ERM)

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(g(x_i, w), y_i)$$

Вопрос: Как называется такой способ приближения интеграла? Монте-Карло сэмплирование.

Математическая постановка задачи обучения: сравнение

$$\min_{w \in \mathbb{R}^d} f(w) := \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\ell(g(x,w),y)\right] \text{ vs } \min_{w \in \mathbb{R}^d} \hat{f}(w) := \frac{1}{n}\sum_{i=1}^{n} \ell(g(x_i,w),y_i)$$

Хотим решить левую задачу (найти вектор w^*, дающий минимум f), а решаем правую (находим $\hat{w}^*$ для $\hat{f}$). **Вопрос:** Насколько близки эти задачи? Какие фактор могут влиять на близость?

Математическая постановка задачи обучения: сравнение

$$\min_{w \in \mathbb{R}^d} f(w) := \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\ell(g(x,w),y)\right] \text{ vs } \min_{w \in \mathbb{R}^d} \hat{f}(w) := \frac{1}{n}\sum_{i=1}^{n} \ell(g(x_i,w),y_i)$$

Хотим решить левую задачу (найти вектор w^*, дающий минимум f), а решаем правую (находим $\hat{w}^*$ для $\hat{f}$). **Вопрос:** Насколько близки эти задачи? Какие фактор могут влиять на близость?

Теорема

Если функция $\ell(g(x,w),y)$ выпукла M-Липшицева по w, тогда с вероятностью хотя бы $1-\delta$

$$f(\hat{w}^*) - f(w^*) = O\left(\sqrt{\frac{M^2 d \ln(n) \ln(d/\delta)}{n}}\right).$$

Переобучение

- Из-за того, что мы решили не совсем ту задачу и нашли $\hat{w}^*$ вместо w^*, и по факту «подстроились» под имеющиеся данные, может случиться эффект **переобучения**.

Переобучение

- Из-за того, что мы решили не совсем ту задачу и нашли $\hat{w}^*$ вместо w^*, и по факту «подстроились» под имеющиеся данные, может случиться эффект **переобучения**.

Переобучение (overfitting).

Не имея проблем с известными данными, модель работает плохо на уже новых данных, которые пришли из $\mathcal{D}$.

Переобучение

- Из-за того, что мы решили не совсем ту задачу и нашли $\hat{w}^*$ вместо w^*, и по факту «подстроились» под имеющиеся данные, может случиться эффект **переобучения**.

Переобучение (overfitting).

Не имея проблем с известными данными, модель работает плохо на уже новых данных, которые пришли из $\mathcal{D}$.

- Эффект переобучения может произойти прежде всего из-за двух вещей:
  1. нехватки данных/большого отличия обучающей и тестовой выборок,

Переобучение

- Из-за того, что мы решили не совсем ту задачу и нашли $\hat{w}^*$ вместо w^*, и по факту «подстроились» под имеющиеся данные, может случиться эффект **переобучения**.

Переобучение (overfitting).

Не имея проблем с известными данными, модель работает плохо на уже новых данных, которые пришли из $\mathcal{D}$.

- Эффект переобучения может произойти прежде всего из-за двух вещей:
  1. нехватки данных/большого отличия обучающей и тестовой выборок,
  2. слишком большой и сложной модели

Переобучение

- Из-за того, что мы решили не совсем ту задачу и нашли $\hat{w}^*$ вместо w^*, и по факту «подстроились» под имеющиеся данные, может случиться эффект **переобучения**.

Переобучение (overfitting).

Не имея проблем с известными данными, модель работает плохо на уже новых данных, которые пришли из $\mathcal{D}$.

- Эффект переобучения может произойти прежде всего из-за двух вещей:
  1. нехватки данных/большого отличия обучающей и тестовой выборок,
  2. слишком большой и сложной модели – удивительно, оказывается, что данных может быть нормальное количество, но из-за того, что используется слишком подробная модель, она воспринимает природу $\mathcal{D}$ слишком буквально.

Примеры переобучения

Представим задачу регрессии на данных, имеющих распределение вида параболы с некоторым гауссовским шумом. Есть три модели – линейная регрессия, полиномиальная со степенью 2 и полиномиальная высших порядков (степень больше 2). **Вопрос:** какая модель покажет себя лучше?

Примеры переобучения

Представим задачу регрессии на данных, имеющих распределение вида параболы с некоторым гауссовским шумом. Есть три модели – линейная регрессия, полиномиальная со степенью 2 и полиномиальная высших порядков (степень больше 2). **Вопрос:** какая модель покажет себя лучше?

1. Первая модель не даст должного качества, так как линией невозможно хорошо приблизить параболу;

Примеры переобучения

Представим задачу регрессии на данных, имеющих распределение вида параболы с некоторым гауссовским шумом. Есть три модели – линейная регрессия, полиномиальная со степенью 2 и полиномиальная высших порядков (степень больше 2). **Вопрос:** какая модель покажет себя лучше?

1. Первая модель не даст должного качества, так как линией невозможно хорошо приблизить параболу;
2. Вторая модель прекрасно покажет себя, так как полностью соответствует данным;

Примеры переобучения

Представим задачу регрессии на данных, имеющих распределение вида параболы с некоторым гауссовским шумом. Есть три модели – линейная регрессия, полиномиальная со степенью 2 и полиномиальная высших порядков (степень больше 2). **Вопрос:** какая модель покажет себя лучше?

1. Первая модель не даст должного качества, так как линией невозможно хорошо приблизить параболу;
2. Вторая модель прекрасно покажет себя, так как полностью соответствует данным;
3. Третья модель слишком сложна – вместо параболы она будет выдавать полиномы большей степени проходящих через каждую точку данных, что не соответствует сути данных.

Борьба с переобучением: big data

Бороться с переобучением можно на различных этапах:

Борьба с переобучением: big data

Бороться с переобучением можно на различных этапах:

- Очевидный способ: чем больше данных, тем лучшего качество обучения можно добиться. **Вопрос:** всегда ли увеличение данных ведет к улучшению качества работы модели?

Борьба с переобучением: big data

Бороться с переобучением можно на различных этапах:

- Очевидный способ: чем больше данных, тем лучшего качество обучения можно добиться. **Вопрос:** всегда ли увеличение данных ведет к улучшению качества работы модели? Нет
- Переобучение (overfitting) Модель может быть слишком простой, чтобы хорошо «подстроится» под природу, как бы много данных у нас ни было. Не зря нейросети становятся все больше и больше (или зря?)

Борьба с переобучением: регуляризация

- Другой вариант – модифицировать задачу оптимизации: поменять функцию потерь или добавить ограничения на w:

$$\min_{w \in \mathbb{R}^d} \left[\hat{f}(w) + \frac{\lambda}{2} \|w\|_2^2 \right] \quad \text{или} \quad \min_{\|w\|_2 \leq D} \hat{f}(w)$$

Вопрос: Что дают такие изменения?

Борьба с переобучением: регуляризация

- Другой вариант – модифицировать задачу оптимизации: поменять функцию потерь или добавить ограничения на w:

$$\min_{w \in \mathbb{R}^d} \left[\hat{f}(w) + \frac{\lambda}{2} \|w\|_2^2 \right] \quad \text{или} \quad \min_{\|w\|_2 \leq D} \hat{f}(w)$$

Вопрос: Что дают такие изменения? И в этом и в другом случае мы хотим, чтобы w не уходили далеко от 0. Первый трюк называется **регуляризацией**.

Регуляризация

Цель регуляризации часто заключается не только в "удержании" весов модели w. Другие виды регуляризации, например, использование ℓ_1-нормы может обеспечить разряженность итогового вектора весов. Почему и зачем – не сегодня.

Борьба с переобучением: валидация

- Задача поиска $\hat{w}^*$ для $\hat{f}$ решается численно, т.е. мы итеративно приближаемся к решению. Можно просто не доходить до $\hat{w}^*$, а остановиться раньше (early stopping). **Вопрос:** но как понять уже пора или нет?

Борьба с переобучением: валидация

Разделим имеющуюся выборку на две части: тренировочную (train) и валидационную (validation). Будем решать задачу минимизации только на *train* части:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n^{\text{train}}} \sum_{i=1}^{n^{\text{train}}} \ell(g(x_i^{\text{train}}, w), y_i^{\text{train}})$$

Борьба с переобучением: валидация

Разделим имеющуюся выборку на две части: тренировочную (train) и валидационную (validation). Будем решать задачу минимизации только на *train* части:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n^{\text{train}}} \sum_{i=1}^{n^{\text{train}}} \ell(g(x_i^{\text{train}}, w), y_i^{\text{train}})$$

- Запустим процесс решения задачи минимизации, стартуем из w^0.

Борьба с переобучением: валидация

Разделим имеющуюся выборку на две части: тренировочную (train) и валидационную (validation). Будем решать задачу минимизации только на *train* части:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n^{\text{train}}} \sum_{i=1}^{n^{\text{train}}} \ell(g(x_i^{\text{train}}, w), y_i^{\text{train}})$$

- Запустим процесс решения задачи минимизации, стартуем из w^0.
- Знаем, что $w^k \to \hat{w}^*_{\text{train}}$ для $k \in \mathbb{N}$. **Вопрос:** Что мы можем сказать про качество решения $\hat{w}^*_{\text{train}}$?

Борьба с переобучением: валидация

Разделим имеющуюся выборку на две части: тренировочную (train) и валидационную (validation). Будем решать задачу минимизации только на *train* части:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n^{\text{train}}} \sum_{i=1}^{n^{\text{train}}} \ell(g(x_i^{\text{train}}, w), y_i^{\text{train}})$$

- Запустим процесс решения задачи минимизации, стартуем из w^0.
- Знаем, что $w^k \to \hat{w}^*_{\text{train}}$ для $k \in \mathbb{N}$. **Вопрос:** Что мы можем сказать про качество решения $\hat{w}^*_{\text{train}}$? В общем случае оно даже хуже, чем $\hat{w}^*$.

Борьба с переобучением: валидация

Разделим имеющуюся выборку на две части: тренировочную (train) и валидационную (validation). Будем решать задачу минимизации только на *train* части:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n^{\text{train}}} \sum_{i=1}^{n^{\text{train}}} \ell(g(x_i^{\text{train}}, w), y_i^{\text{train}})$$

- Запустим процесс решения задачи минимизации, стартуем из w^0.
- Знаем, что $w^k \to \hat{w}^*_{\text{train}}$ для $k \in \mathbb{N}$. **Вопрос:** Что мы можем сказать про качество решения $\hat{w}^*_{\text{train}}$? В общем случае оно даже хуже, чем $\hat{w}^*$.
- Но в силу того, что w^0 плохое решение, как для f, так и для $\hat{f}$ и $\hat{f}^{\text{train}}$. Поэтому до какого-то момента процесс оптимизации w^k приближается к w^*.

Борьба с переобучением: валидация

Разделим имеющуюся выборку на две части: тренировочную (train) и валидационную (validation). Будем решать задачу минимизации только на *train* части:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n^{\text{train}}} \sum_{i=1}^{n^{\text{train}}} \ell(g(x_i^{\text{train}}, w), y_i^{\text{train}})$$

- Запустим процесс решения задачи минимизации, стартуем из w^0.
- Знаем, что $w^k \to \hat{w}^*_{\text{train}}$ для $k \in \mathbb{N}$. **Вопрос:** Что мы можем сказать про качество решения $\hat{w}^*_{\text{train}}$? В общем случае оно даже хуже, чем $\hat{w}^*$.
- Но в силу того, что w^0 плохое решение, как для f, так и для $\hat{f}$ и $\hat{f}^{\text{train}}$. Поэтому до какого-то момента процесс оптимизации w^k приближается к w^*. **Вопрос:** Как словить момент для остановки процесса (помните про валидационную выборку)?

Борьба с переобучение: валидация

- При минимизации функции f^{train} (обучении) мы не видели валидационную выборку. Естественная идея, проверить на ней качество текущего решения w^k, предполагая, что валидационная выборка – это новые данные пришедшие из неизвестного распределения $\mathcal{D}$, которое мы и хотим найти. **Вопрос:** Как проверить?

Борьба с переобучение: валидация

- При минимизации функции f^{train} (обучении) мы не видели валидационную выборку. Естественная идея, проверить на ней качество текущего решения w^k, предполагая, что валидационная выборка – это новые данные пришедшие из неизвестного распределения $\mathcal{D}$, которое мы и хотим найти. **Вопрос:** Как проверить? Например, измерить потери на валидационной выборке:

$$\hat{f}^{\text{val}}(w^k) := \frac{1}{n^{\text{val}}} \sum_{i=1}^{n^{\text{val}}} \ell(g(x_i^{\text{val}}, w^k), y_i^{\text{val}}),$$

Борьба с переобучение: валидация

- При минимизации функции f^{train} (обучении) мы не видели валидационную выборку. Естественная идея, проверить на ней качество текущего решения w^k, предполагая, что валидационная выборка – это новые данные пришедшие из неизвестного распределения $\mathcal{D}$, которое мы и хотим найти. **Вопрос:** Как проверить? Например, измерить потери на валидационной выборке:

$$\hat{f}^{\text{val}}(w^k) := \frac{1}{n^{\text{val}}} \sum_{i=1}^{n^{\text{val}}} \ell(g(x_i^{\text{val}}, w^k), y_i^{\text{val}}),$$

и словить некоторый момент времени k^*, что потери на валидации начинают расти: $f^{\text{val}}(w^{k^*-1}) \geq f^{\text{val}}(w^{k^*}) < f^{\text{val}}(w^{k^*+1})$.

Основные требования к модели

- Качество – высокие показатели метрик в задаче (например, точность предсказаний на тестовой выборке);

Основные требования к модели

- Качество – высокие показатели метрик в задаче (например, точность предсказаний на тестовой выборке);
- Эффективность – время обучения/инференса;

Основные требования к модели

- Качество – высокие показатели метрик в задаче (например, точность предсказаний на тестовой выборке);
- Эффективность – время обучения/инференса;
- Робастность – устойчивость к шуму в данных;

Основные требования к модели

- Качество – высокие показатели метрик в задаче (например, точность предсказаний на тестовой выборке);
- Эффективность – время обучения/инференса;
- Робастность – устойчивость к шуму в данных;
- Масштабируемость – при увеличении объема данных поведение модели не изменится;

Основные требования к модели

- Качество – высокие показатели метрик в задаче (например, точность предсказаний на тестовой выборке);
- Эффективность – время обучения/инференса;
- Робастность – устойчивость к шуму в данных;
- Масштабируемость – при увеличении объема данных поведение модели не изменится;
- Интерпретируемость – объясняемость результатов модели;

Основные требования к модели

- Качество – высокие показатели метрик в задаче (например, точность предсказаний на тестовой выборке);
- Эффективность – время обучения/инференса;
- Робастность – устойчивость к шуму в данных;
- Масштабируемость – при увеличении объема данных поведение модели не изменится;
- Интерпретируемость – объясняемость результатов модели;
- Компактность – затраты на хранение модели.

Другие виды обучения

1. Обучение без учителя (Un-/Self-supervised learning) – нужно «понять» структуру пространства $\mathcal{X}$;

Другие виды обучения

1. Обучение без учителя (Un-/Self-supervised learning) – нужно «понять» структуру пространства $\mathcal{X}$;
2. Обучение с частично размеченными данными (Semi-supervised learning) – $X = \{(x_1, y_1), \ldots, (x_k, y_k), x_{k+1}, \ldots, x_n\}$;

Другие виды обучения

1. Обучение без учителя (Un-/Self-supervised learning) – нужно «понять» структуру пространства $\mathcal{X}$;
2. Обучение с частично размеченными данными (Semi-supervised learning) – $X = \{(x_1, y_1), \ldots, (x_k, y_k), x_{k+1}, \ldots, x_n\}$;
3. Обучение с подкреплением (Reinforcemnet learning) – агент взаимодействует со средой и обучается посредством получения наград за свои действия;

Другие виды обучения

1. Обучение без учителя (Un-/Self-supervised learning) – нужно «понять» структуру пространства $\mathcal{X}$;
2. Обучение с частично размеченными данными (Semi-supervised learning) – $X = \{(x_1, y_1), \ldots, (x_k, y_k), x_{k+1}, \ldots, x_n\}$;
3. Обучение с подкреплением (Reinforcemnet learning) – агент взаимодействует со средой и обучается посредством получения наград за свои действия;
4. Онлайн обучение (Online learning) – в каждый конкретный момент доступна лишь небольшая выборка объектов;

Другие виды обучения

1. Обучение без учителя (Un-/Self-supervised learning) – нужно «понять» структуру пространства $\mathcal{X}$;
2. Обучение с частично размеченными данными (Semi-supervised learning) – $X = \{(x_1, y_1), \ldots, (x_k, y_k), x_{k+1}, \ldots, x_n\}$;
3. Обучение с подкреплением (Reinforcemnet learning) – агент взаимодействует со средой и обучается посредством получения наград за свои действия;
4. Онлайн обучение (Online learning) – в каждый конкретный момент доступна лишь небольшая выборка объектов;
5. Обучение с переносом (Transfer learning) – решение новых задач с помощью решений старых.