Машинное обучение.

Алексей Колесов

Белорусский государственный университет

4 сентября 2019 г.

Содержание

Домашние задания

Домашнее задание выдаётся после каждой лекции; по три балла за каждое задание

- работа сдана в течение 7 дней — коэффициент **2**
- работа сдана в течение 14 дней — коэффициент **1**
- работа не сдана в течение 14 дней — коэффициент **0**

Кроме того, в некоторых домашних работах будут встречаться задания повышенной сложности, которые могут повысить вашу оценку (при условии, что домашняя работа сдана за две недели).

Семинарские занятия

- консультируемся насчёт домашних заданий
- разбираем домашние задания
- решаем задания по теме лекции на дополнительные баллы

Экзамен

Пересдача автоматом:

- получить ноль более, чем по двум домашкам
- получить баллов меньше, чем $3H$, где H – количество домашек

Экзамен автоматом:

- сданы все домашки
- количество баллов больше $4.5H$

Получение баллов

- работа на лекциях
- решение задач на семинарах
- решение домашних заданий
- дополнительные задания (по желанию)
- контрольные мероприятия

Дополнительные замечания

- посещение лекций и семинаров не влияет на оценивание
- компьютер вам понадобится лишь для решения домашних заданий
- лекционный материал зачастую сложней семинарских и домашних заданий
- в этом курсе очень строгое отношение к дедлайнам!

Содержание

Что такое машинное обучение

Том Митчелл

A computer program is said to learn from experience **E**, with respect to some task **T**, and some performance measure **P**, if its performance on **T** as measured by **P** improves with experience **E**.

Примеры обучения в окружающем мире

- крысы *учатся* избегать отравленные приманки (если было плохо после маленького кусочка, больше такое не есть)
- эксперимент с голубями[1] показывает, что они суеверны (готовы делать что угодно, если это приносит еду)
- фундаментальный вопрос: как нам различить полезное обучение от суеверия
- эксперименты с крысами[2] показывают, что они не видят связи между вкусом еды и ударами тока; или между звуком и плохим самочувствием

[1] http://psychclassics.yorku.ca/Skinner/Pigeon
[2] Garcia & Koelling 1996

Некоторые понятия

Обобщающая способность — качество программы показывать хорошее качество на примерах, которые она не видела раньше

Inductive bias — набор предположений (априорных знаний), который используется для предсказания неизвестных значений

Для успешного обучения использование априорных знаний неизбежно (No Free Lunch theorem).

Зачем нужно машинное обучение

- задачи, которые сложно запрограммировать
    - сложноформализуемые задачи (например, распознавание символов, речи, вождение автомобиля)
    - задачи неподвластные человеку (анализ астрономических данных, ранжирование веб-страниц)
- задачи, для которых нужна адаптация

Типы обучения

- с учителем и без учителя
- активное и пассивное
- онлайн и оффлайн

Типовые задачи машинного обучения

- классификация
- регрессия
- ранжирование
- кластеризация
- уменьшение размерности

Связь с другими дисциплинами

- оптимизация
  - в ML применяются методы оптимизации
  - задача оптимизации найти экстремум известной функции; задача ML — построить решение, которое хорошо обобщается
- статистика
  - обе дисциплины — инструменты над теорией вероятности; в ML много статистических методов
  - задача статистики — проверить гипотезу; задача ML — выдвинуть гипотезу

О чём этот курс

- математические основы машинного обучения
- обзор алгоритмов машинного обучения
- анализ алгоритмов с точки зрения их производительности и гарантий

Что в этом курсе не будет

- обзор библиотек машинного обучения
- советов, как победить на Kaggle

Содержание

Формальная модель обучения

- X — множество объектов (**domain set**); характеризуется набором признаков (**features**)
- Y — множество меток (**label set**); например, вкусный/не вкусный $\{0, 1\}$
- $S = ((x_1, y_1), \ldots, (x_m, y_m))$, $x_i \in X$, $y_i \in Y$ — тренировочные данные, тренировочная выборка (**training set**); $S|_x = (x_1, \ldots, x_m)$ — набор объектов в тренировочной выборке
- $h : X \to Y$ — ответ алгоритма, гипотеза, решающее правило (**hypothesis, predictor**)
- A — алгоритм машинного обучения: $A(S) = h$

Простая модель генерации данных

- Предположим, что существует распределение вероятности D над множеством объектов X
- Каждый объект тренировочной выборки $x_i \in S|_x$ выбирается независимо из D (предположение о независимости и одинаковой распределённости)
- Существует функция $f : X \to Y$, $y_i = f(x_i)\ \forall i$ (предположение о детерминированности среды)
- Ошибка классификации: $L_{D,f}(h) = \underset{x \sim D}{\mathbb{P}}[h(x) \neq f(x)]$ (**generalization error**, **true risk**)

Информация, доступная алгоритму

- алгоритм знает только про выборку S
- у него нет никаких знаний про D (тут отличие от математической статистики)
- у него нет никаких знаний про f (более того, ровно её мы и хотим найти)

Минимизация эмпирического риска

Модель: алгоритм принимает S, полученный из распределения D и размеченный функцией f. Его задача найти гипотезу $h_S : X \to Y$, который минимизирует ошибку $L_{D,f}(h_S)$ по отношению к **неизвестным** D и f.

- D и f неизвестны $\Rightarrow L_{D,f}(h_s)$
- давайте использовать ошибку на тренировочной выборке (**empirical risk, empirical error**):

$$L_S(h) = \frac{|\{i \in [m] : h(x_i) \neq y_i\}|}{m}$$

Минимизация эмпирического риска — парадигма обучения, заключающаяся в выборе гипотезы, минимизирующей ошибку на тренировочной выборке

Переобучение

На практике, минимизация эмпирического риска зачастую приводит к переобучению (**overfitting**).

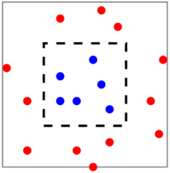

Рис.: Пример распределения драников

Пример (*вероятно, плохой*) ERM-гипотезы:

$$h_S(x) = \begin{cases} y_i & \text{если } \exists i \in [m] \text{ такой что } x_i = x \\ 0 & \text{иначе} \end{cases}$$

ERM with inductive bias

- ERM-правило приводит к переобучению
- Вместо того, чтоб не использовать его, найдём случаи, когда это правило работает достаточно хорошо
- Хороший способ — ограничить набор гипотез
- H — семейство гипотез из X в Y;

$$\mathrm{ERM}_H(S) \in \operatorname*{argmin}_{h \in H} L_S(h)$$

- Один из важных вопросов машинного обучения: «для каких H ERM_H не переобучается»

ERM для конечного набора гипотез

- Пусть набор гипотез H — конечен
- Сделаем следующее предположение

Предположение реализуемости — найдётся такая гипотеза $h^* \in H$, что $L_{D,f}(h^*) = 0$

- Предположение значит, что (с вероятностью 1) найдётся гипотеза $h \in H$, что $L_S(h) = 0$
- Но нам интересен $L_D(h)$, а не $L_S(h)$
- Предположим, что $S \sim D^m$ (предположение о независимости и одинаковой распределённости)

ERM для конечного набора гипотез

- S — случайная величина $\Rightarrow h_S = ERM_H(S)$ — случайная величина $\Rightarrow L_{D,f}(h_S)$ — случайная величина
- Нереалистично предполагать, что мы всегда сможем получить хороший $L_{D,f}(h_S)$ (нам может очень не повезти с выборкой)
- Кроме того, даже если с выборкой повезло, её конечность всё равно означает то, что полностью D мы оценить не сможем

ERM для конечного набора гипотез

Чтоб оценить эти замечания, введём два параметра:

- δ — вероятность того, что нам попадётся плохая (нерепрезентативная) выборка (число $1-\delta$ называется **confidence parameter**)
- ϵ — максимальная ошибка, которую мы разрешаем иметь нашему классификатору на D (**accuracy**)
- $L_{D,f}(h_S) > \epsilon$ — неуспех алгоритма

ERM для конечного набора гипотез

Мы хотим оценить сверху следующую вероятность:

$$D^m(\{S|_x : L_{D,f}(h_S) > e\})$$

Введём набор «плохих» гипотез:

$$H_B = \{h \in H : L_{D,f}(h) > e\}$$

И набор «опасных» тренировочных выборок:

$$M = \{S|_x : \exists h \in H_B, L_S(h) = 0\}$$

Из-за предположения о реализуемости:

$$\{S|_x : L_{D,f}(h_S) > e\} \subseteq M$$

ERM для конечного набора гипотез

Мы имеем:

$$M = \{S|_x : \exists h \in H_B, L_S(h) = 0\}$$

$$\{S|_x : L_{D,f}(h_S) > e\} \subseteq M$$

Перепишем M:

$$M = \bigcup_{h \in H_B} \{S|_x : L_S(h) = 0\}$$

Отсюда:

$$D^m(\{S|_x : L_{D,f}(h_S) > e\}) \leqslant D^m\left(\bigcup_{h \in H_B} \{S|_x : L_S(h) = 0\}\right)$$

ERM для конечного набора гипотез

Union bound

Для любых двух событий и любого распределения вероятностей:

$$\mathbb{P}(A \cup B) \leqslant \mathbb{P}(A) + \mathbb{P}(B)$$

$$D^m \left(\bigcup_{h \in H_B} \{S|_x : L_S(h) = 0\} \right) \leqslant \sum_{h \in H_B} D^m \left(S|_x : L_S(h) = 0\right)$$

$$D^m(\{S|_x : L_{D,f}(h_S) > e\}) \leqslant \sum_{h \in H_B} D^m \left(S|_x : L_S(h) = 0\right)$$

ERM для конечного набора гипотез

Оценим каждое слагаемое в предыдущей сумме

$$D^m(S|_x : L_S(h) = 0) = D^m(S|_x : \forall i, h(x_i) = f(x_i)) =$$
$$= \prod_{i=1}^{m} D(\{x : h(x) = f(x)\})$$

Так как $h \in H_B$:

$$D(\{x : h(x) = f(x)\} = 1 - L_{D,f}(h) \leqslant 1 - \epsilon$$

Так как $1 - \epsilon \leqslant e^{-\epsilon}$:

$$D^m(S|_x : L_S(h) = 0) \leqslant (1 - \epsilon)^m \leqslant e^{-\epsilon m}$$

ERM для конечного набора гипотез

Соберём вместе:

$$D^m(\{S|_x : L_{D,f}(h_S) > e\}) \leqslant \sum_{h \in H_B} D^m(S|_x : L_S(h) = 0)$$
$$\leqslant |H_B| e^{-\epsilon m} \leqslant |H| e^{-\epsilon m}$$

ERM для конечного набора гипотез

Пусть H — конечный набор гипотез. Возьмём произвольные $\delta \in (0,1)$ и $\epsilon > 0$ и число m, такое что:

$$m \geqslant \frac{\log\left(|H|/\delta\right)}{\epsilon}$$

Тогда для любой функции f, для любого распределения D (с выполненным предположением о реализуемости) с вероятностью как минимум $1-\delta$ ERM-гипотеза h_S от выборки S размера m, порождённой независимо распределением D и размеченной функцией f, выполняется:

$$L_{D,f}(h_S) \leqslant \epsilon$$

Probably approximately correct learnability

Класс гипотез H называют **вероятно приблизительно верно изучаемым** (probably approximately correct learnable) если существует такая функция $m_H : (0,1)^2 \to \mathbb{N}$ и алгоритм, такой что

- для любых $\epsilon, \delta \in (0,1)$
- для любого распределения D над X
- для любой функции $f : X \to \{0,1\}$

если выполняется предположение о реализуемости, то если мы выполним алгоритм на выборке из $m \geqslant m_H(\epsilon, \delta)$ независимых одинаково распределённых элементов из D и размеченных f, то алгоритм вернёт гипотезу $h \in H$ такую, что с вероятностью как минимум $1 - \delta$, выполняется $L_{D,f}(h) \leqslant \epsilon$

Выборочная сложность

Функция m_H называется выборочной сложностью (**sample complexity**)
Зависит от:

- ϵ — accuracy
- δ — confidence
- H (например, для конечных классов мы видели, что зависит от $\log |H|$)

Любой конечный класс является PAC-изучаемым с выборочной сложностью

$$m_H(\epsilon, \delta) \leqslant \left\lceil \frac{\log (|H|/\delta)}{\epsilon} \right\rceil$$

Обобщение PAC-learnable model

- Избавимся от предположения о реализуемости
- Добавим в модель не только бинарную классификацию

Избавление от предположения о реализуемости

- Пусть теперь D — распределение над $X \times Y$, например $D(x, y) = D_x(x) \cdot D((x, y)|x)$
- True risk:

$$L_D(h) = \underset{(x,y)\sim D}{\mathbb{P}}[h(x) \neq y] = D(\{(x, y) : h(x) \neq y\})$$

- Эмпирический риск:

$$L_S(h) = \frac{|\{i \in [m] : h(x_i) \neq y_i\}|}{m}$$

Оптимальный байесовский классификатор

Наша цель — найти гипотезу, у которой $L_D(h)$ будет (вероятно приблизительно) низким.

Оптимальный байесовский классификатор

Классификатор:

$$f_D(x) = \operatorname*{argmax}_y D(x, y)$$

называется **оптимальным байесовским классификатором**

Agnostic PAC-learnable

Класс гипотез H называют **агностически вероятно приблизительно верно изучаемым** (agnostic PAC-learnable) если существует такая функция $m_H : (0,1)^2 \to \mathbb{N}$ и алгоритм, такой что

- для любых $\epsilon, \delta \in (0,1)$
- для любого распределения D над $X \times Y$
- ~~для любой функции~~ $\cancel{f : X \to \{0,1\}}$

~~если выполняется предположение о реализуемости, то~~ если мы выполним алгоритм на выборке из $m \geqslant m_H(\epsilon, \delta)$ независимых одинаково распределённых элементов из D ~~и размеченных~~ $\cancel{f}$, то алгоритм вернёт гипотезу $h \in H$ такую, что с вероятностью как минимум $1 - \delta$, выполняется $L_D(h) \leqslant \min_{h' \in H} L_D(h') + \epsilon$

Обобщённые функции потерь

- Пусть дан набор гипотез H и некий домен Z. Пусть есть функция $l : H \times Z \to \mathbb{R}_+$; такую функцию будем называть *функцией потерь* (**loss function**)
- True risk: $L_D(h) = \mathop{\mathbb{E}}_{z \sim D}[l(h,z)]$
- Empirical risk: $L_S(h) = \frac{1}{m} \sum_{i=1}^{m} l(h, z_i)$
- **0-1 loss** $l_{01}(h,(x,y)) = [h(x) = y]$
- **Square loss** $l_{sq}(h,(x,y)) = (h(x) - y)^2$

Agnostic PAC-leanable for generalized loss functions

Класс гипотез H называют **агностически вероятно приблизительно верно изучаемым** (agnostic PAC-learnable) по отношению к множеству Z и функции потерь $l : H \times Z \to \mathbb{R}_+$, если существует такая функция $m_H : (0,1)^2 \to \mathbb{N}$ и алгоритм, такой что

- для любых $\epsilon, \delta \in (0,1)$
- для любого распределения D над Z

если мы выполним алгоритм на выборке из $m \geqslant m_H(\epsilon, \delta)$ независимых одинаково распределённых элементов из D, то алгоритм вернёт гипотезу $h \in H$ такую, что с вероятностью как минимум $1 - \delta$, выполняется $L_D(h) \leqslant \min_{h' \in H} L_D(h') + \epsilon$, где $L_D(h) = \mathop{\mathbb{E}}_{z \sim D}[l(h,z)]$

Содержание

Идея обучения через равномерную сходимость

- ERM-алгоритм выбирает самую лучшую гипотезу по отношению к тренировочной выборке
- True risk для такой гипотезы может оказаться гораздо выше, чем empirical risk
- **Идея**: давайте найдём такие условия, что у любой гипотезы из класса true risk несильно отличается от empirical risk

Тренировочная выборка S называется ϵ-**репрезентативной** (по отношению к домену Z, классу гипотез H, функции потерь l и распределению D), если

$$\forall h \in H, |L_D(h) - L_S(h)| \leqslant \epsilon$$

Лемма о ERM в случае $\frac{\epsilon}{2}$-репрезентативной выборки

Лемма о ERM в случае $\frac{\epsilon}{2}$-репрезентативной выборки

Пусть выборка S является $\frac{\epsilon}{2}$-репрезентативной. Тогда если h_S — ERM-гипотеза, то выполняется:

$$L_D(h_S) \leqslant \min_{h \in H} L_D(h) + \epsilon$$

Равномерная сходимость

Класс гипотез H обладает свойством **равномерной сходимости (uniform convergence)** (по отношению к домену Z и функции потерь l), если существует такая функция $m_H^{UC} : (0,1)^2 \to \mathbb{N}$, что для любых δ, ϵ из $(0,1)$ и любого распределения D над Z, если выборка S состоит из $m \geqslant m_H^{UC}(\epsilon, \delta)$ объектов, выбранных из D независимо, то с вероятностью как минимум $1 - \delta$, S является ϵ-репрезентативной выборкой.

Если класс H обладает свойством равномерной сходимости с выборочной сложностью m_H^{UC}, то этот же класс является агностически PAC-изучаемым с выборочной сложностью $m_H(\epsilon, \delta) \leqslant m_H^{UC}(\epsilon/2, \delta)$. Более того, в этом случае ERM-алгоритм — успешный PAC-learner

Доказательство PAC-learnability для конечных классов

Хотим показать, что:

$$D^m(\{S : \forall h \in H, |L_S(h) - L_D(h)| \leqslant \epsilon\}) \geqslant 1 - \delta$$

План следующий:

1. Применим union bound
2. Оценим каждый элемент суммы

Доказательство PAC-learnability для конечных классов: шаг 1

Хотим:

$$D^m(\{S : \forall h \in H, |L_S(h) - L_D(h)| \leqslant \epsilon\}) \geqslant 1 - \delta$$

То же самое, что:

$$D^m(\{S : \exists h \in H, |L_S(h) - L_D(h)| > \epsilon\}) < \delta$$

Опять перепишем:

$$\{S : \exists h \in H, |L_S(h) - L_D(h)| > \epsilon\} = \bigcup_{h \in H} \{S : |L_S(h) - L_D(h)| > \epsilon\}$$

Доказательство PAC-learnability для конечных классов: шаг 1

Хотим:

$$D^m(\{S : \forall h \in H, |L_S(h) - L_D(h)| \leqslant \epsilon\}) \geqslant 1 - \delta$$

Получили:

$$D^m(\{S : \exists h \in H, |L_S(h) - L_D(h)| > \epsilon\}) \leqslant \sum_{h \in H} D^m(\{S : |L_S(h) - L_D(h)| > \epsilon\})$$

Доказательство PAC-learnability для конечных классов: шаг 2

- Надо показать, что для любой выбранной h величина $|L_D(h) - L_S(h)|$ с высокой вероятностью небольшая относительно выбора S
- Напомним, что $L_D(h) = \underset{z \sim D}{\mathbb{E}}[l(h, z)]$, а $L_S(h) = \frac{1}{m} \sum_{i=1}^{m} l(h, z_i)$
- $\mathbb{E}[L_S(h)] = L_D(h)$ — очень полезное наблюдение!
- А значит, $|L_D(h) - L_S(h)|$ — отклонение случайной величины $L_S(h)$

Доказательство PAC-learnability для конечных классов: шаг 2

- Нам нужно оценить, насколько величина сконцентрирована относительного своего среднего
- Закон больших чисел говорит, что выборочное среднее стремится к своему среднему, но это лишь асимптотический закон

Неравенство Хёфдинга

Пусть $\theta_1, \ldots, \theta_m$ — последовательность независимых одинаково распределённых на $[a; b]$ случайных величин с $\mathbb{E}[\theta_i] = \mu$. Тогда для любого $\epsilon > 0$:

$$\mathbb{P}\left[\left|\frac{1}{m}\sum_{i=1}^{m}\theta_i - \mu\right| > \epsilon\right] \leqslant 2\exp(-2m\epsilon^2/(b-a)^2)$$

Доказательство PAC-learnability для конечных классов: шаг 2

Пусть $\theta_i = l(h, z_i)$; тогда $L_S(h) = \frac{1}{m}\sum_{i=1}^{m}\theta_i$, $L_D(h) = \mu$

Также, предположим, что l имеет значения только в $[0;1]$.

$$
\begin{aligned}
D^m\left(\{S : |L_S(h) - L_D(h)| > \epsilon\}\right) = \\
= \mathbb{P}\left[\left|\frac{1}{m}\sum_{i=1}^{m}\theta_i - \mu\right| > \epsilon\right] \leqslant \\
\leqslant 2\exp(-2m\epsilon^2)
\end{aligned}
$$

Доказательство PAC-learnability для конечных классов

А значит:

$$D^m(\{S : \exists h \in H, |L_S(h) - L_D(h)| > \epsilon\}) \leqslant 2|H| \exp(-2m\epsilon^2)$$

Таким образом, если взять

$$m \geqslant \frac{log(2|H|/\delta)}{2\epsilon^2}$$

То:

$$D^m(\{S : \exists h \in H, |L_S(h) - L_D(h)| > \epsilon\}) \leqslant \delta$$

Содержание

Итоги

- Сегодня мы сделали обзор основных понятий машинного обучения
- Построили основную модель обучения — agnostic PAC-learnability
- Разработали инструмент для доказательства agnostic PAC-learnability — uniform convergence
- Доказали, что ERM-алгоритм является agnostic PAC-learnable для конечного класса гипотез

Литература

- Shai Shalev-Shwartz and Shai Ben-David — Understanding Machine Learning: From theory to algorithms (главы 2-4)
- Mehryar Mohri — Foundations of Machine Learning (Lection 01) — `http://www.cs.nyu.edu/~mohri/mls/ml_introduction.pdf`
- Andrew Ng — Coursera Machine Learning course — lection 1