



Московский энергетический институт

Кафедра Вычислительных машин систем и сетей

КУРС ПРОБЛЕМЫ ОРГАНИЗАЦИИ ВЫЧИСЛЕНИЙ

Лекция №2 на тему :

«Введение в машинное обучение»

Москва 2018 г.

Задача классификации (обучение с y)

Задача восстановления зависимости $y: X \rightarrow Y$, $|Y| < \infty$
по точкам обучающей выборки (x_i, y_i) , $i = 1, \dots, \ell$.

Дано: векторы $x_i = (x_i^1, \dots, x_i^n)$ — объекты обучающей выборки,
 $y_i = y(x_i)$ — классификации, ответы учителя, $i = 1, \dots, \ell$:

$$\begin{pmatrix} x_1^1 & \dots & x_1^n \\ \dots & \dots & \dots \\ x_\ell^1 & \dots & x_\ell^n \end{pmatrix} \xrightarrow{y^*} \begin{pmatrix} y_1 \\ \dots \\ y_\ell \end{pmatrix}$$

Найти: функцию $a(x)$, способную классифицировать объекты
произвольной тестовой выборки $\tilde{x}_i = (\tilde{x}_i^1, \dots, \tilde{x}_i^n)$, $i = 1, \dots, k$:

$$\begin{pmatrix} \tilde{x}_1^1 & \dots & \tilde{x}_1^n \\ \dots & \dots & \dots \\ \tilde{x}_k^1 & \dots & \tilde{x}_k^n \end{pmatrix} \xrightarrow{a?} \begin{pmatrix} a(\tilde{x}_1) \\ \dots \\ a(\tilde{x}_k) \end{pmatrix}$$

Определение бинарного решающего дерева

Бинарное решающее дерево — алгоритм классификации $a(x)$, задающийся бинарным деревом:

1) $\forall v \in V_{\text{внутр}} \rightarrow$ предикат $\beta_v : X \rightarrow \{0, 1\}$, $\beta_v \in \mathcal{B}$,

2) $\forall v \in V_{\text{лист}} \rightarrow$ имя класса $c_v \in Y$,

где \mathcal{B} — множество бинарных признаков или предикатов (например, вида $\beta(x) = [x^j \geq \theta_j]$, $x^j \in \mathbb{R}$)

1: $v := v_0$;

2: **пока** $v \in V_{\text{внутр}}$

3: **если** $\beta_v(x) = 1$ **то**

4: переход вправо: $v := R_v$;

5: **иначе**

6: переход влево: $v := L_v$;

7: **вернуть** c_v .



Определение бинарного решающего дерева

Бинарное решающее дерево — алгоритм классификации $a(x)$, задающийся бинарным деревом:

1) $\forall v \in V_{\text{внутр}} \rightarrow$ предикат $\beta_v : X \rightarrow \{0, 1\}$, $\beta_v \in \mathcal{B}$,

2) $\forall v \in V_{\text{лист}} \rightarrow$ имя класса $c_v \in Y$,

где \mathcal{B} — множество бинарных признаков или предикатов (например, вида $\beta(x) = [x^j \geq \theta_j]$, $x^j \in \mathbb{R}$)

1: $v := v_0$;

2: **пока** $v \in V_{\text{внутр}}$

3: **если** $\beta_v(x) = 1$ **то**

4: переход вправо: $v := R_v$;

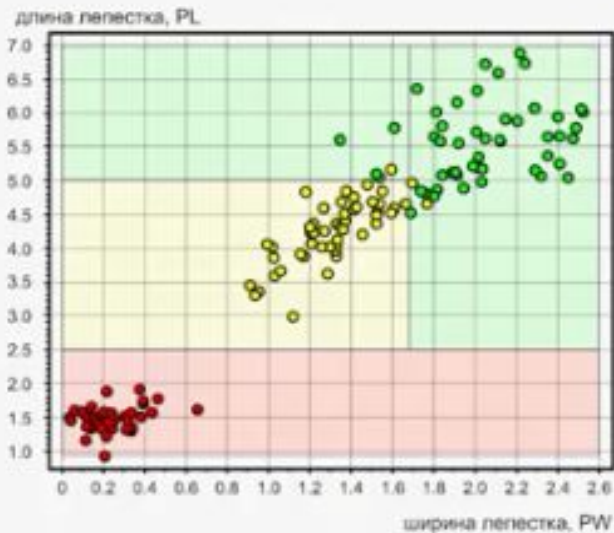
5: **иначе**

6: переход влево: $v := L_v$;

7: **вернуть** c_v .



Решающее дерево → покрывающий набор конъюнкций



setosa	$r_1(x) = [PL \leq 2.5]$
virginica	$r_2(x) = [PL > 2.5] \wedge [PW > 1.68]$
virginica	$r_3(x) = [PL > 5] \wedge [PW \leq 1.68]$
versicolor	$r_4(x) = [PL > 2.5] \wedge [PL \leq 5] \wedge [PW < 1.68]$

Жадный алгоритм построения дерева ID3

- 1: **ПРОЦЕДУРА** LearnID3 ($U \subseteq X^l$);
- 2: **если** все объекты из U лежат в одном классе $c \in Y$ **то**
- 3: **вернуть** новый лист v , $c_v := c$;
- 4: **найти предикат с максимальной информативностью:**
 $\beta := \arg \max_{\beta \in \mathcal{B}} I(\beta, U)$;
- 5: разбить выборку на две части $U = U_0 \sqcup U_1$ по предикату β :
 $U_0 := \{x \in U : \beta(x) = 0\}$;
 $U_1 := \{x \in U : \beta(x) = 1\}$;
- 6: **если** $U_0 = \emptyset$ или $U_1 = \emptyset$ **то**
- 7: **вернуть** новый лист v , $c_v := \text{Мажоритарный класс}(U)$;
- 8: создать новую внутреннюю вершину v : $\beta_v := \beta$;
построить левое поддереву: $L_v := \text{LearnID3}(U_0)$;
построить правое поддереву: $R_v := \text{LearnID3}(U_1)$;
- 9: **вернуть** v ;

Варианты критериев ветвления

1. Критерий Джини:

$$I(\beta, X^\ell) = \#\{(x_i, x_j): y_i = y_j \text{ и } \beta(x_i) = \beta(x_j)\}.$$

2. D -критерий В.И.Донского:

$$I(\beta, X^\ell) = \#\{(x_i, x_j): y_i \neq y_j \text{ и } \beta(x_i) \neq \beta(x_j)\}.$$

3. Энтропийный критерий:

$$I(\beta, X^\ell) = \sum_{c \in Y} h\left(\frac{P_c}{\ell}\right) - \frac{p}{\ell} h\left(\frac{p_c}{p}\right) - \frac{\ell - p}{\ell} h\left(\frac{P_c - p_c}{\ell - p}\right),$$

где $h(z) \equiv -z \log_2 z$,

$$P_c(X^\ell) = \#\{x_i: y_i = c\},$$

$$p_c(X^\ell) = \#\{x_i: y_i = c \text{ и } \beta(x_i) = 1\},$$

$$p(X^\ell) = \#\{x_i: \beta(x_i) = 1\}.$$

Решающие деревья ID3: достоинства

Достоинства:

- › Интерпретируемость и простота классификации.
- › Гибкость: можно варьировать множество \mathcal{B} .
- › Допустимы разнотипные данные и данные с пропусками.
- › Трудоёмкость линейна по длине выборки $O(|\mathcal{B}|hl)$.
- › Не бывает отказов от классификации.

Недостатки:

- › Жадный ID3 переусложняет структуру дерева, и, как следствие, сильно переобучается.
- › Фрагментация выборки: чем дальше v от корня, тем меньше статистическая надёжность выбора β_v, c_v .
- › Высокая чувствительность к шуму, к составу выборки, к критерию информативности.

Усечение дерева (pruning). Алгоритм C4.5

X^k — независимая контрольная выборка, $k \approx 0.5\ell$.

- 1: для всех $v \in V_{\text{внутр}}$
- 2: $S_v :=$ подмножество объектов X^k , дошедших до v ;
- 3: если $S_v = \emptyset$ то
- 4: вернуть новый лист v , $c_v :=$ Мажоритарный класс(U);
- 5: число ошибок при классификации S_v четырьмя способами:
 - $r(v)$ — поддеревом, растущим из вершины v ;
 - $r_L(v)$ — поддеревом левой дочерней вершины L_v ;
 - $r_R(v)$ — поддеревом правой дочерней вершины R_v ;
 - $r_c(v)$ — к классу $c \in Y$.
- 6: в зависимости от того, какое из них минимально:
 - сохранить поддерево v ;
 - заменить поддерево v поддеревом L_v ;
 - заменить поддерево v поддеревом R_v ;
 - заменить поддерево v листом, $c_v := \arg \min_{c \in Y} r_c(v)$.